Closed-Loop Nominal and Abort Atmospheric Ascent Guidance for Rocket-Powered Launch Vehicles

A Dissertation
By
Greg A. Dukeman

Submitted to
School of Aerospace Engineering

In Partial Fulfillment of the
Requirements for the Degree
Doctor of Philosophy in Aerospace Engineering

Georgia Institute of Technology

May 2005

Closed-Loop Nominal and Abort Atmospheric Ascent Guidance for Rocket-Powered Launch Vehicles

Approved by:

Dr. Anthony J. Calise,Advisor School of Aerospace Engineering *Georgia Institute of Technology*		Dr. Aldo Ferri School of Mechanical Engineering *Georgia Institute of Technology*
Dr. Dewey H. Hodges School of Aerospace Engineering *Georgia Institute of Technology*		Dr. John M. Hanson *NASA Marshall Flight Center*
Dr. Panagiotas Tsiotras School of Aerospace Engineering *Georgia Institute of Technology*		Date Approved: January 11, 2005

ACKNOWLEDGEMENT

Many people contributed to the research and results reported in this thesis. My colleagues in the Guidance, Navigation and Control Systems Group at the NASA Marshall Space Flight Center contributed in substantial ways. Dr. John Hanson has been a constant source of encouragement, advice and motivation to me for many years. Thanks to my friend Jim McCarter for developing one of the country's most capable trajectory simulations and helping with simulation questions. A sincere thank you goes to my thesis advisor, Dr. Anthony Calise, for his assistance and patience and for demanding nothing less than high-quality work. Thanks to each of my Final Committee members who collectively provided a very thorough review of the thesis: Dr. Dewey H. Hodges and Dr. Panagiotis Tsiotras of the School of Aerospace Engineering, Dr. Aldo Ferri of the School of Mechanical Engineering, and Dr. John M. Hanson of the NASA Marshall Space Flight Center. Thanks to NASA and the Marshall Space Flight Center for providing so many great opportunities for continuing engineering education. Much of this research was supported in particular by the Next Generation Launch Technology project at NASA-Marshall – I express my gratitude.

I would like to thank my wife Margaret for her love and support over the years and for tolerating many late evenings. Reducing my hours somewhat sounds like a winning proposition to me. Thanks to my kids, Anton and Alex, just for being the beautiful human beings that you are. The world is a better place with you in it. Thanks to my parents, Bud and Eileen (deceased) for instilling in me a love of learning.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

Acronyms

ATDS	Abort To Down range Site
COV	Calculus Of Variations
IGM	Iterative Guidance Mode
MAVERIC	Marshall Aerospace VEhicle Representation In C trajectory simulator
MECO	Main Engine CutOff
MSFC	Marshall Space Flight Center
PEG	Powered Explicit Guidance
psf	pounds per square foot
POST	Program to Optimize Simulated Trajectories
RTLS	Return To Launch Site
TAEM	Terminal Area Energy Management

Symbols

A	aerodynamic axial force magnitude
a	speed of sound
a	transversality vector
C_A	coefficient of axial force
C_{A0}, C_{A1}, C_{A2}	axial force coefficient polynomial coefficients
C_N	coefficient of normal force
C_{N0}, C_{N1}, C_{N2}	normal force coefficient polynomial coefficients

c	rocket exhaust velocity (= g_0 Isp)
g	gravity acceleration vector
g_0	sea-level gravity acceleration
H	Hamiltonian function
H_{NT}	thrust-independent portion of the Hamiltonian
h	vehicle angular momentum vector
M	Mach number
m	vehicle mass
N	aerodynamic normal force magnitude
S	aerodynamic reference area, switching function
r	vehicle position vector
r	vehicle position vector magnitude
q	vehicle dynamic pressure
$q\alpha$	product of dynamic pressure and angle of attack
$q\beta$	product of dynamic pressure and angle of sideslip
T	vehicle thrust magnitude
t_f	predicted cutoff time from guidance
$\mathbf{u_p}$	unit vector along (Earth) spin axis
v	vehicle velocity vector
v	vehicle velocity vector magnitude
x	column vector containing position and velocity vectors
$\mathbf{x_b}$, $\mathbf{y_b}$, $\mathbf{z_b}$	x-, y-, and z-body axes unit vectors
α	vehicle angle of attack
β	vehicle angle of sideslip
γ	vehicle flight path angle

η	fractional throttle level
δ	angle between velocity costate vector and $\mathbf{x_b}$ vector
ϕ	angle between position vector and velocity costate vector, or the scalar part of the performance index
ϕ_0	angle between Earth-relative velocity vector and velocity costate vector
$\Delta T(h)$	actual thrust minus vacuum thrust, a function of altitude
$\boldsymbol{\lambda}$	column vector containing position and velocity costate vectors
$\boldsymbol{\lambda_r}$	position vector costate
λ_{ry}, λ_{rz}	components of position vector costate
$\boldsymbol{\lambda_v}$	velocity vector costate
ρ	atmospheric density
σ_{dir}	variable set to +1 or –1, specifying heads-up or heads-down flight, resp.
ω_0	parameter associated with invocation of linear gravity field assumption
ω	rotation rate about the (Earth) polar axis

Subscripts

0	initial, or reference value
d	desired value
I	vector or quantity relative to the inertial frame
i	ith value or occurrence
r	vector or quantity relative to the Earth-fixed frame
vac	in vacuum, e.g., vacuum thrust

SUMMARY

An advanced ascent guidance algorithm for rocket-powered launch vehicles is developed. The ascent guidance function is responsible for commanding attitude, throttle and setting during the powered ascent phase of flight so that the vehicle attains target cutoff conditions in a near-optimal manner while satisfying path constraints such as maximum allowed bending moment and maximum allowed axial acceleration. This algorithm cyclically solves the calculus-of-variations two-point boundary-value problem starting at vertical rise completion through orbit insertion. This is different from traditional ascent guidance algorithms which operate in an open-loop mode until the high dynamic pressure portion of the trajectory is over, at which time there is a switch to a closed loop guidance mode that operates under the assumption of negligible aerodynamic forces.

The main contribution of this research is an algorithm of the predictor-corrector type wherein the state/costate system is propagated with known (navigated) initial state and guessed initial costate to predict the state/costate at engine cutoff. The initial costate guess is corrected, using a multi-dimensional Newton's method, based on errors in the terminal state constraints and the transversality conditions. Path constraints are enforced within the propagation process. A modified multiple shooting method is shown to be a very effective numerical technique for this application. Results for a single-stage-to-orbit launch vehicle are given. In addition, the formulation for the free final time multi-arc trajectory optimization problem is given. Results for a two-stage launch vehicle burn-coast-burn ascent to orbit in a closed-loop guidance mode are shown. An abort to landing site formulation of the algorithm and numerical results are presented. A technique for numerically treating the transversality conditions is discussed that eliminates part of the analytical and coding burden associated with optimal control theory.

CHAPTER 1

INTRODUCTION

1.1 Background

NASA, through a program known as the Space Launch Initiative [1] (also, Second Generation Reusable Launch Vehicle program), has set for itself the goals of significantly increasing the safety and reliability of the second generation of launch vehicles by two orders of magnitude (loss of crew to 1 in 10,000 flights), while reducing the launch costs by an order of magnitude (to $1,000/pound payload). The second generation program office is working with industry, academia, and government organizations to develop and increase the technology readiness levels of several technology areas including airframes, propulsion, flight demonstrations, flight mechanics, integrated vehicle health management, operations, and vehicle subsystems. Within the flight mechanics technology area, advanced ascent, on-orbit, and entry guidance, navigation and control algorithms are being developed and tested. In order for the flight mechanics discipline to contribute to the goals stated above, it is important that these algorithms be highly robust and adaptive to in-flight failures such as actuator/aerosurface failures and partial engine loss.

Marshall Space Flight Center's Advanced Guidance and Control project [2], begun in 1999 and still in progress, is developing and testing robust ascent, entry, and TAEM guidance and control algorithms. The goal of the project is to advance the algorithm designs to the point that

they can successfully steer and control the vehicle after any failure from which an intact recovery is physically possible. If an actuator fails, for example during entry, an advanced attitude control system would detect this (whether implicitly or explicitly) and utilize other control surfaces to the extent possible to compensate. If the failure were to result in insufficient control authority to maintain stable tracking of the guidance commands, then an advanced (entry) guidance system would detect this (implicitly or explicitly) and adjust its commands accordingly, e.g., it may limit the commanded bank angle rates to, say, 3 degrees per second instead of 5 degrees per second. Similarly, if an engine fails during ascent flight, the ascent guidance algorithm must be able to compute a feasible abort trajectory when one exists. Candidate abort scenarios could include press-to-Main Engine Cutoff (MECO)--that is, continue the ascent with the same orbit target values as before--abort to orbit with alternate orbit targets, abort to a downrange landing site and abort to launch site.

Guidance, as defined in this thesis, is that part of a launch vehicle's onboard flight software that is responsible for controlling the vehicle's translational motion so that the mission objectives, such as insertion into a 160 n.m. circular orbit, are met while satisfying constraints, such as maximum bending moment or maximum allowed axial acceleration. The translational motion is determined by the forces acting on the vehicle which in turn are primarily a function of the vehicle attitude and throttle setting. Hence, the guidance function computes the desired attitude and throttle values needed to complete the mission. In order to accurately maintain trajectory control, guidance is typically executed at a pre-determined rate, e.g., 1 Hz, throughout the mission, using updated state estimates from the navigation system. The desired values, or commands, are sent from guidance to the attitude control system which is responsible for implementing the guidance commands by commanding control effectors such as aerodynamic control surfaces, engine gimbal angles, or differential throttling. Because the rotational dynamics

are faster than the translational dynamics, the attitude control function typically cycles faster than guidance, e.g., at a rate of 25 or 50 Hz.

Traditional ascent guidance algorithms have relied heavily on attitude profiles of pre-computed trajectories which are stored onboard to be used by ascent guidance in open-loop mode, that is, generation of desired attitudes via stored attitude profiles. The traditional approach is to operate in an open-loop mode during the (early) high dynamic pressure portion of flight and then, based on a pre-determined time or event, to switch to a closed-loop in vacuo guidance scheme which operates on the premise that aerodynamic forces can be neglected. The open-loop mode typically makes use of pre-loaded tables of optimal steering angles versus time or speed [3,4]. The closed-loop logic is based on explicit formulas and simplified dynamics that result in a semi-analytical solution for the optimal steering angles [3,5,6]. This partitioning of the flight into distinct phases was necessary primarily due to computer throughput and memory limitations as well as the lack of availability of advanced algorithms that take aerodynamic forces into consideration. Introduction of aerodynamic forces into the problem formulation makes the problem much more sensitive and computationally-intensive. Simple, well-understood formulas do not exist as they do in the in vacuo case, making the complete liftoff-to-burnout optimization problem difficult to solve reliably in real-time. However, since it is highly probable that computer power will continue to increase, it will be feasible to use more sophisticated, more robust onboard algorithms capable of increasing the reliability and safety, and of reducing the operational costs, in the next generation of launch vehicles.

The objective of this thesis is to develop closed-loop launch vehicle atmospheric ascent guidance for both nominal and abort mission scenarios. It is required that such an algorithm be as computationally efficient as possible to satisfy onboard computer throughput requirements realizing that onboard computer capabilities (e.g., clock speed) typically lag their ground-based counterparts. The algorithm must be reliable with respect to its ability to compute feasible

solutions when they exist. In the case of an abort scenario (e.g., partial engine failure), whenever it is physically possible to recover the vehicle intact, it is desired that the guidance algorithm be capable of generating an acceptable recovery trajectory which can be used to steer the vehicle to an engine cutoff state that is compatible with a safe abort to orbit or to a landing site.

1.2 Previous Research

Several ascent guidance methods have been developed and studied over the past 40 years. Most treat only the exo-atmospheric phase of flight while more recently, methods that also treat the endo-atmospheric phase have been developed. Research has also been performed on trajectory tracking methods wherein it is assumed that an optimal trajectory is already available.

The Saturn/Apollo ascent guidance algorithm (known as the Iterative Guidance Mode, or IGM) [5,6] was based on calculus of variations results [7]. The basic result, under the assumption of a constant gravity field, is that the tangent of the optimum pitch angle is a linear function of time. The formulation contained several approximations (e.g., small pitch and yaw angle maneuvers, average gravity model, etc.), to help simplify the code and to make it compatible with very stringent computer memory and speed limitations. Despite all the approximations, it provides the same (to within a few pounds) performance (payload mass to orbit) as in vacuo guidance algorithms that use a minimum of simplifications provided that the ascent burn extends over no more than, say, 30 degrees of arc. Brown et al. [8,9] developed an in vacuo guidance algorithm which solves, with a minimum of simplifying assumptions, the nonlinear two-point boundary value problem and demonstrated its effectiveness for performing both ascent and orbital powered maneuvers. Burrows and McDaniel [10] developed an in vacuo algorithm which included an analytical solution for the thrust integrals and Taylor series expansions for the gravity along with the assumption of linear thrust vector steering (as defined in Chapter 3). Their work

demonstrated that linear steering is nearly optimal even for relatively long burn arcs, say, on the order of 90 degrees. McHenry et al. [3] developed what could be described as a 'vector' version of IGM, the result of which is the space shuttle vacuum ascent and orbit maneuvering guidance algorithm, PEG, which uses linear vector steering combined with numerical integration to accurately obtain the gravity effects over long burn arcs. More recently, Sinha and Shrivastava [11] developed a vacuum guidance algorithm using recursive series for the thrust integrals and Encke's method for the gravity integrals and applied it to sun-synchronous and geostationary transfer orbit missions.

Lu [12] developed a nonlinear trajectory tracking guidance algorithm that tracks a pre-computed trajectory while guaranteeing satisfaction of angle of attack and normal force path constraints. Seywald [13] used a neighboring optimal control based feedback law for guidance of a two-stage advanced launch system vehicle. Although these (trajectory-tracking) methods are fairly robust and perform well under moderately off-nominal conditions, they are not expected to be robust enough to adapt to large dispersions like partial engine losses.

Some of the earliest work into extending the capabilities of ascent guidance algorithms so that they would be effective just after liftoff was done by Brown et al. [14] and by Ingram [15]. In Ref. [14] a linearized aerodynamics model was used to obtain the optimal control (thrust direction) from the optimality condition in closed form. Curve fits for the lift and drag coefficients and atmospheric density, pressure and speed of sound were used in the guidance formulation to reduce the computational burden imposed by aerodynamics modeling. The conventional shooting method combined with a homotopy procedure was used to solve the two-point boundary-value problem. The easily obtained vacuum solution was first computed, from which a homotopy procedure was used to re-introduce increments of the atmospheric effects. Despite the use of homotopy to reduce sensitivity to the initial costate guesses, reliable convergence was not always attained. Kelly developed a similar algorithm formulation in [16]

with similar convergence difficulties reported, despite attempted homotopy procedures, due to inclusion of atmospheric terms. Much of Kelly's research and ideas for atmospheric guidance formulation were inspired by work done as early as 1964 by Dickey, Gottlieb, and Battelle [17,18,19]. Bradt et al. [20] used a formulation similar to [14] and added a penalty function to reduce bending moment loads. Cramer et al. [21] used a nonlinear programming approach to guidance and take advantage of measured day-of-launch winds in the guidance to provide load relief. Hanson et al. developed and tested several atmospheric ascent guidance options in [22]. Leung and Calise used a perturbation approach in [23]. Calise and Melamed [24] used a hybrid collocation approach and demonstrate reliable convergence in dispersed guided trajectory simulations. In [25], Gath and Calise extended previous work to normal force and angle of attack path constraints and optimization of burn-coast-burn sequences.

Most of the afore-mentioned research was done under the assumption of bang-bang control wherein the engines are operated either at maximum thrust or turned off. Under some circumstances, it is more fuel-optimal to fly a singular arc, i.e., a trajectory phase wherein the engines are throttled to intermediate values of throttle. Bryson and Ho [7] discussed the Goddard sounding rocket problem which, in general, involves a maximum thrust arc, followed by an intermediate arc, followed by a coast arc. Ross [26] derived the expressions for obtaining optimal angle of attack for a non-gimballed vehicle flying a singular arc. Casalino [27] derived the expressions for optimum throttle for a gimbaled vehicle flying a singular arc within an aerocruise maneuver. Calise and Brandt [28] showed that, for the problem of steerable rocket flight, like that considered herein, singular arcs are not feasible; hence, singular arcs will be excluded from consideration.

Brown, et al, [8] developed necessary conditions for the multi-burn trajectory optimization problem. They used single shooting and a modified Newton's method to solve for the initial costates and the switching times. They performed closed-loop guidance by cyclically re-

optimizing the trajectory. Their multi-burn work treated only orbital transfers for orbiting vehicles. Jezewski [42] treated the multi-burn trajectory optimization problem using a linear gravity assumption with parameter optimization to numerically solve for the costates. His work focused on trajectory generation for mission planning and analysis. Gath and Calise [25] generated open-loop burn-coast-burn ascent-to-orbit trajectories including atmospheric effects using a hybrid method. Each of these studies used a set of necessary conditions that is not sufficiently restrictive in the sense that the switching function is not guaranteed to vanish at switching points.

Return to launch site trajectory shaping and guidance techniques for the space shuttle have been studied extensively. McHenry et al. [3] discussed the space shuttle RTLS guidance techniques. Bown [29] and Carter and Bown [30] discussed RTLS abort trajectory shaping techniques for the space shuttle while Sponagle et al. [31] treated guidance techniques for RTLS contingency aborts during first stage space shuttle flight. Kishi et al. [32] discussed the post-powered flight guidance techniques for space shuttle RTLS abort. Dutton [33] and Stanley and Powell [34] discussed abort trajectory generation and capabilities for the HL-20 Personnel Launch System vehicle and for a single-stage launch vehicle, respectively. Chuang and Ledsinger [35] presented an RTLS guidance scheme based on neighboring optimal control. Abort trajectories were computed using a hybrid optimization method in Calise and Brandt [28].

1.3 Main Contributions

This thesis describes in detail an atmospheric guidance algorithm for computing near-optimal thrust direction histories from which real-time vehicle attitude commands are generated. The focus is on the ascent phase of rocket-powered launch vehicles although there is nothing in the

formulation to preclude it from being used equally effectively as an on-orbit powered flight guidance algorithm. The application of a modified multiple shooting method is used and is shown to provide quick and reliable convergence. Emphasis is placed on developing and testing algorithm formulations in a closed-loop (guidance) mode to provide early design feedback on considerations such as computational burden, minimum time-to-go, mis-modeling effects, and convergence reliability in the face of vehicle and environmental dispersions. For the problem of guidance algorithm formulation, a judicious blend of mathematical optimality, algorithm simplicity, convergence reliability and speed is indicated. Accommodation of path constraints, e.g., $q\alpha$, $q\beta$, α, β, and q is treated. Day-of-launch winds are used by the guidance to support the latter.

Chapter 2 describes the atmospheric equations of motion and derives the Euler-Lagrange differential equations as well as the optimality condition that is used to determine the optimal thrust direction. The equations of motion allow for the practical situation of non-zero thrust component along the *z*-body axis, a departure from previous research which assumed thrust is perfectly aligned with the *x*-body axis. The numerical method of multiple shooting as applied to the ascent problem is presented. The equations are derived for the case of wings-level (bank equal to 0 or 180 deg) flight, in contrast to previous research [25,41] which uses a zero sideslip formulation. The advantage of the wings-level formulation is that no roll attitude maneuvering is required as contrasted with the zero sideslip formulation which calls for roll maneuvering to keep the wind-relative velocity in the *x*-*z* plane. Chapter 3 describes a vacuum guidance formulation that is applicable to those portions of flight when the aerodynamic forces are small compared to the thrust forces. An analytical solution to the thrust integrals is combined with "average gravity" integrals resulting in a near-optimal guidance solution applicable to both ascent guidance and orbital transfer guidance. The formulation avoids numerical quadrature without relying on small angle approximations. Chapter 4 presents numerical results using the methods of Chapters 2 and

3. Guided sub-orbital as well as ascent-to-orbit trajectories for a single-stage-to-orbit launch vehicle are shown. Chapter 5 discusses various methods of treating the transversality conditions including a numerical method that dispenses with the constraint-dependent analyses commonly used to obtain a reduced set of transversality conditions. This technique allows for a guidance protocol wherein individual terminal state constraints are selected from a menu to achieve a particular mission objective. A combinatorially large number of terminal state constraint sets are thus available without any extra mathematical derivation (i.e., elimination of constant Lagrange parameters [28]) or computer coding.

Chapter 6 presents a free final time multi-arc trajectory optimization formulation in which the interior point switching conditions are derived. A new necessary condition guarantees that the switching function vanishes at interior switching points. This is in contrast to previous work [8,25,42] that merely required the switching function to take on equal, possibly non-zero, values at each interior point. The multiple shooting method for solving the multi-arc optimization problem is discussed and open-loop numerical results for a two-stage-to-orbit launch vehicle are provided. Chapter 7 discusses details pertinent to implementing a closed-loop guidance that uses the burn-coast-burn formulation. Closed-loop guidance results for the case of optimal burn-coast-burn ascent are given. Several options for the terminal state constraint sets for such missions are presented and compared in terms of guidance performance. Chapter 8 presents an abort to landing site trajectory generation procedure. The procedure is guaranteed to converge to a solution whenever the vehicle has enough fuel to reach a candidate landing site. The trajectory generation formulation of Calise and Brandt [28] results in a nonlinear multi-dimensional root-finding problem that has no convergence guarantee. The abort trajectory generation problem is highly constrained by fuel depletion and terminal state constraints and is particularly sensitive to initial guesses and initial conditions. A method that guarantees convergence is hence very desirable, if not required, for the onboard guidance application. In Chapter 9, a method is

described for implementing closed-loop guidance that uses the abort trajectory solution of Chapter 8 as part of a reference trajectory tracking scheme. Guided abort to launch site and abort to down range trajectories are presented. The research in [28] does not treat the abort guidance problem. Chapter 10 gives conclusions and suggestions for further research.

Appendix A lists miscellaneous relations that are useful in ascent guidance algorithms, such as analytical equations for computing mass and throttle values as a function of time during axial acceleration-limited flight phases. Appendix B provides a mathematical proof that the necessary condition that the Hamiltonian at the final time must vanish, $H(t_f) = 0$, is an ignorable constraint for the multi-arc trajectory optimization problem and, hence, can be replaced with a simplified constraint. Appendix C states and proves a theorem that provides conditions under which the thrust-independent part of the Hamiltonian automatically vanishes at the final time of an extremal trajectory. Appendix D describes the efficient and accurate methods used in this research to analytically calculate the partial derivatives required in the build-up of the Jacobian matrix. Appendix E lists vehicle properties for the two launch vehicles used to test the formulations developed in this thesis. The first is the single-stage-to-orbit X-33 flight demonstrator, while the second is a generic two-stage-to-orbit vehicle. Appendix F describes a new method for treating, within the trajectory generation process, launch vehicles with gimbaled engines wherein the thrust is not necessarily directed along the vehicle's x-body axis, referred to as axially asymmetric. This effect is important to take into account for those asymmetric vehicles where the net thrust force is several degrees offset from the x-body axis. Otherwise, the guidance commands will be erroneously offset by the same angle, resulting in large trajectory deviations and mission threatening performance.

CHAPTER 2

ATMOSPHERIC ASCENT GUIDANCE FORMULATION

In this chapter, expressions and relations required for solving, via optimal control theory, the ascent trajectory optimization problem are developed. The equations of motion for a launch vehicle in atmospheric flight are discussed. The problem is a highly constrained problem involving both terminal state constraints and path constraints. Optimal control theory is applied, resulting in a nonlinear two-point boundary value problem. The case of single burn optimization is treated here while multi-arc (e.g., burn-coast-burn) trajectory optimization is treated in Chapter 6. While the formulation of this chapter could in principle be used, along with appropriate terminal state constraints, to generate abort trajectories [28], a specialized technique is presented in Chapter 8 that increases the chances of finding these highly constrained trajectories.

Appendix F discusses modeling techniques and incorporation of guidance logic necessary to handle gimballed-engine vehicles that result in z-component thrust. A wings-level formulation is presented here, as contrasted with a zero sideslip formulation presented in Calise and Gath [25] and Lu et al. [41]. A multiple shooting method is discussed here as the numerical means of solving the two-point boundary value problem. This is in contrast to the hybrid method described in Refs. [23,24,25,28] which uses fixed point iteration to solve for atmosphere-related parameters and the finite differences method of Lu et al. [41]. Chapter 4 discusses numerical results using the formulation given here and Chapter 3. In particular, guided trajectories are presented with

closed-loop guidance implemented via cyclical re-optimization of the trajectory using the latest navigation data.

2.1 Atmospheric Equations of Motion

The equations of motion for a thrusting rocket in atmospheric flight are written as:

$$\begin{aligned}
\dot{\mathbf{r}} &= \mathbf{v} \\
\dot{\mathbf{v}} &= \mathbf{g} + \frac{(T - A)\mathbf{x_b} - N\mathbf{z_b}}{m} \\
\dot{m} &= -\frac{\eta T_{vac}}{c}
\end{aligned} \tag{2-1}$$

where:

$$\begin{aligned}
T &= T(h) = T_{vac} + \Delta T(h) \\
A &= A(h, \alpha, M, v) = \tfrac{1}{2}\rho v_r^2 S C_A \equiv q S C_A \\
N &= N(h, \alpha, M, v) = \tfrac{1}{2}\rho v_r^2 S C_N \equiv q S C_N \\
\rho &= \rho(h) = \rho_0 \exp(\rho_1 h + \rho_2 h^2 + \rho_3 h^3) \\
a &= a(h) = a_0 + a_1 h + a_2 h^2 + a_3 h^3 \\
C_A &= C_A(\alpha, M) = C_{A_0}(M) + C_{A_1}(M)\alpha + C_{A_2}(M)\alpha^2 \\
C_N &= C_N(\alpha, M) = C_{N_0}(M) + C_{N_1}(M)\alpha + C_{N_2}(M)\alpha^2 \\
v_r^2 &= (\mathbf{v} - \mathbf{w} - (\omega \mathbf{u_p} \times \mathbf{r}))^T (\mathbf{v} - \mathbf{w} - (\omega \mathbf{u_p} \times \mathbf{r})) \\
\mathbf{g} &= \mathbf{g}(\mathbf{r}) = -\frac{\mu}{r^3}\mathbf{r} \\
\mathbf{w} &= \mathbf{w}(h)
\end{aligned} \tag{2-2}$$

and **r** is the vehicle's position vector from the center of the Earth to the vehicle center of gravity, **v** is the vehicle's inertial velocity vector, T is the net thrust taking into account any back pressure terms, A and N are the axial and normal aerodynamic forces, resp., ρ is the ambient density, a is the ambient speed of sound, C_A and C_N are the axial and normal aerodynamic force coefficients, v_r

is the vehicle's air-relative velocity magnitude, **g** is the gravity acceleration vector taken here to be a central gravity field, and **w** is the velocity of the air relative to a frame fixed in the rotating Earth. Note that bold-face quantities throughout this thesis denote column vectors. In the guidance formulation, the atmospheric density and speed of sound models are of the form given in (2-2) which are least squares curve fits of US 1976 standard atmosphere computed off-line. The fitted data match the US 1976 data to within 5 percent up to 70 km. The aerodynamic force coefficients are modeled by least squares quadratic (or higher) polynomials in angle of attack and by cubic splines in Mach number. The mixed polynomial-spline model for the aerodynamic force coefficients is suggested by the fact that the coefficients are typically nice, smooth functions in angle of attack but highly nonlinear in Mach, particularly in the transonic region. The wind vector **w** is calculated from a table lookup or cubic spline as a function of altitude and represents a measured wind profile or estimated wind profile on the day of launch. The back pressure terms, ΔT, are also calculated using a cubic spline fitted to roughly ten altitude-delta thrust data pairs. The cubic spline allows for modeling of both nonlinear back pressure terms associated with linear aerospike engines and simpler nearly exponential back pressure terms associated with conventional bell nozzle engines. In all cases, curve fit parameters and cubic splines coefficients are calculated off-line.

The equations of motion given here are valid for the case wherein the thrust vector is aligned with the x-body axis. Appendix F treats the generalized case where the thrust vector has a component along the z-body axis. This situation occurs on rockets where the engines are gimbaled to counter aerodynamic moments or center of gravity offsets.

For now, consider only constant thrust segments or constant thrust acceleration segments so that the vehicle mass, m, is treated as a prescribed function of time, not a state. The velocity vector, **v**, can be taken to be the Earth-relative velocity vector, that is, the velocity of the vehicle's cg in a frame fixed to the rotating Earth, or as the inertial velocity vector, depending on the

context. For high-speed flight (typically occurring 'outside' the atmosphere), inertial velocity is used, whereas for limited duration, low-speed flight, such as sub-orbital missions, it is often sufficient to let the velocity in question be Earth-relative velocity. Note that it has been assumed that the force along the *y*-body axis is small and hence it will be ignored in the equations of motion and consequently in the optimization. In the proceeding developments, the position and velocity vectors are expressed in the guidance coordinate frame as illustrated in Figure 2-1. The guidance coordinate frame is an Earth-centered, right-handed, inertial coordinate system with the *x*-axis aligned with the local vertical and the *z*-axis aligned along the anticipated downrange direction. It is re-defined each guidance cycle via the vehicle's current latitude, longitude and an azimuth angle which approximates the downrange direction of travel[1]. This is a convenient frame to work in because, for example, the initial position vector expressed in the guidance frame has *y*- and *z*-components equal to zero and the *y* components of position and velocity are typically near zero.

[1] The azimuth angle is computed at guidance initiation using target inclination and launch latitude in the spherical trig relation

$$\sin(az) = \frac{\cos(i_{tgt})}{\cos(lat)}$$

.

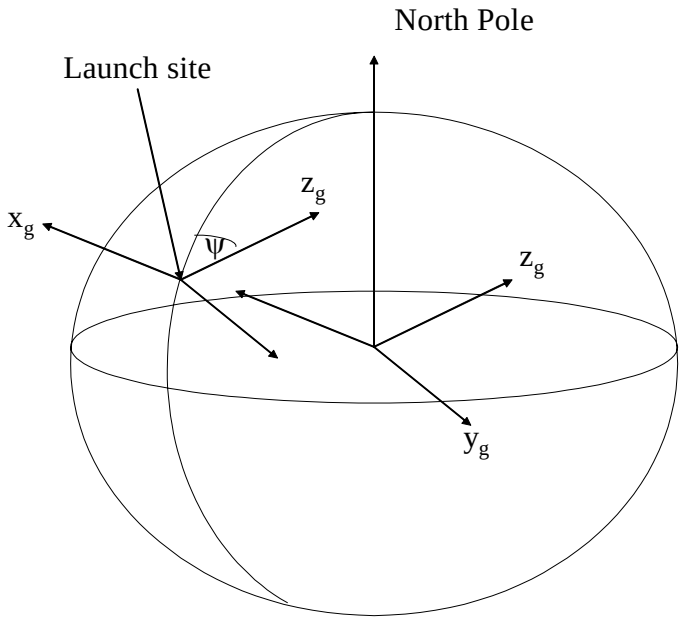

Figure 2-1: Guidance Reference Frame

2.2 Optimization Problem

The quantity to be minimized is propellant usage which is equivalent to maximizing payload to orbit or, equivalently, to maximizing the vehicle's total mass at MECO, $m(t_f)$. This is expressed mathematically as

$$J = \max_{x_b(t)} \int_{t_0}^{t_f} \dot{m} dt = \max_{x_b(t)} \int_{t_0}^{t_f} -\frac{T_{vac}}{c} dt \tag{2-3}$$

For the single burn optimization problem, this is equivalent to the minimum time problem but the form (2-3) is used here to be consistent with the form used later in the treatment of the multi-arc optimization problem. Note that the throttle level, η, is not treated as a control variable here but

rather is taken to be a prescribed function of time or mass. Typically, the throttle is set at its maximum value although it is often beneficial to reduce the throttle setting during the early, high dynamic pressure portion of ascent to help reduce dynamic pressure. Calise and Brandt [28] have shown that intermediate thrust arcs (i.e., singular arcs) are infeasible and need not be considered for atmospheric flight. Burn-coast-burn trajectories are treated later in Chapter 6. Gath and Calise [25] showed that such trajectories can significantly reduce fuel cost for single-stage-to-orbit vehicles.

At the time of engine cutoff, t_f, k terminal state constraints - nonlinear (in general) functions of the states - of the following form are prescribed:

$$\psi_i = \psi_i\left[\mathbf{r}(t_f), \mathbf{v}(t_f)\right] = 0, \quad i = 1,\ldots,k, \quad k \le 6 \tag{2-4}$$

Examples of terminal state constraints include position magnitude, flight path angle, semi-major axis, argument of perigee, inclination and longitude of the ascending node. For a mission where, say, the objective is to transfer to a low Earth orbit and cutoff at a (prescribed) target position magnitude, r_t, target speed, v_t, and target flight path angle, γ_t, the constraint equations (2-4) would be constructed as:

$$\begin{aligned} &\left|\mathbf{r}(t_f)\right| - r_t = 0 \\ &\left|\mathbf{v}(t_f)\right| - v_t = 0 \\ &\mathbf{r}(t_f)^T\,\mathbf{v}(t_f) - r_t v_t \sin(\gamma_t) = 0 \end{aligned} \tag{2-5}$$

where r_t is the target position vector magnitude.

In addition to the terminal state constraints, it is commonly the case that path constraints are imposed on the trajectory. During the atmospheric portion of ascent to orbit flight, the launch vehicle experiences large structural loads due to aerodynamic forces and moments. For winged vehicles like the space shuttle, wing loading is a concern. Bending moments and aerodynamic surface hinge moments are a concern for many launch vehicles. Crew comfort and vehicle structural integrity suggest axial acceleration limits. Some of the most common constraints are maximum allowed dynamic pressure, maximum allowed axial acceleration, maximum allowed product of dynamic pressure and angle of attack, and maximum allowed product of dynamic pressure and angle of sideslip. These can be expressed as

$$\begin{aligned} &\frac{1}{2}\rho v_r^2 - q_{\max} \le 0 \\ &\frac{\eta T_{vac}}{mg} - a_{\max} \le 0 \\ &\left|q\alpha\right| - \left|q\alpha\right|_{\max} \le 0 \\ &\left|q\beta\right| - \left|q\beta\right|_{\max} \le 0 \end{aligned} \tag{2-6}$$

In the performance index (2-3) it is implied that the control to be optimized is the *x*-body axis history. This still theoretically leaves one control history degree of freedom in that the *y*-body axis can take on any arbitrary orientation perpendicular to the *x*-body axis. For practical reasons, this degree of freedom is removed. It is desirable to fly the vehicle in a "heads-up" (bank angle = 0) or heads-down (bank angle = 180 deg) orientation so that telemetry and communications links are more robust, for passenger comfort, and to reduce attitude maneuvering and control effort. This effectively results in angle of attack and sideslip as free controls and angle of bank as a prescribed quantity. In the sequel, this formulation is referred to as the "wings-level" formulation. An alternative formulation that was considered but not developed further here, is to let angle of attack and bank be the free controls and constrain angle of sideslip to zero, i.e., the

"zero-sideslip" formulation [14,25,41]. Given an *x*-body axis unit vector, $\mathbf{x_b}$, and specified bank of 0 or 180 deg, the *y*-body axis unit vector is constructed from

$$\mathbf{y_b} = \pm \frac{\mathbf{x_b} \times \mathbf{r_0}}{\left| \mathbf{x_b} \times \mathbf{r_0} \right|} \tag{2-7}$$

where the positive sign corresponds to bank of 0 and the minus sign corresponds to a specified bank of 180 deg. In (2-7), note that the initial position vector, $\mathbf{r_0}$, (more accurately, the navigated position vector, input each guidance cycle from the navigation subsystem to the guidance subsystem) is used. The choice of $\mathbf{r_0}$, instead of simulated position vector, $\mathbf{r}$, simplifies the necessary conditions of optimality and correspondingly, partial derivatives expressions, because $\mathbf{r_0}$ can be treated as a constant in the trajectory optimization problem. The only potential effect of the simplification is that, for atmospheric flights extending over several degrees of ground track, the bank angle of the modeled vehicle may not correspond exactly to a wings-level attitude. Note that there is potential ill-conditioning in (2-7) whenever the *x*-body axis is nearly aligned with the initial position vector. Fortunately, this near-alignment typically only occurs for a short period just after liftoff.

Contrast the wings-level formulation with the zero-sideslip formulation of [25,41]. To ensure zero sideslip, the *y*-body axis, $\mathbf{y_b}$, is constructed via

$$\mathbf{y_b} = \frac{\mathbf{x_b} \times \mathbf{v}_r}{\left| \mathbf{x_b} \times \mathbf{v}_r \right|} \tag{2-8}$$

in which $\mathbf{x_b}$ and the relative velocity vector, $\mathbf{v_r}$, are often nearly-aligned (alignment corresponds to tangential steering, which approximates optimal steering for many powered flight trajectory problems) and hence, the zero-sideslip formulation can be problematic [41].

The wings-level formulation *z*-body axis unit vector is just the cross product of the *x*- and *y*-body axes

$$\mathbf{z_b} = \mathbf{x_b} \times \mathbf{y_b} = \pm \frac{\mathbf{x_b} \times (\mathbf{x_b} \times \mathbf{r_0})}{|\mathbf{x_b} \times \mathbf{r_0}|} \tag{2-9}$$

Note that in both formulations, the orientation of the *x*-body axis, $\mathbf{x_b}$, is the 'control' to be optimized and $\mathbf{y_b}$ and $\mathbf{z_b}$ are functions of $\mathbf{x_b}$. The zero-sideslip formulation results in a non-zero roll angle and possibly excessive roll maneuvering. It is likely that future launch vehicles will have limited roll control authority and there may be other reasons not to arbitrarily roll the vehicle during launch (e.g., antenna visibility). The heads-up/down option will inherently result in larger angles of sideslip but this can be attenuated fairly easily, if need be, via enforcement of a path constraint on the product of dynamic pressure and sideslip angle, $q\beta$. Worst-case sideslip angles occur on rendezvous missions which can require a lot of non-planar maneuvering, particularly for launches occurring early or late in the launch window. The latter is usually limited to about ten minutes to avoid excessive non-planar maneuvering and the associated excessive fuel usage. In the sequel, the heads-up/heads-down option (which parallels the zero sideslip option) is treated.

The optimization problem can now be stated as follows. Determine the vehicle attitude history that maximizes the final vehicle mass subject to the equations of motion (2-1), the terminal state constraints (2-4), the path constraints (2-6), and the wings-level attitude constraint (2-7).

2.3 Candidate Solution Approaches

Now that the trajectory optimization problem has been defined, candidate solution approaches are considered. Betts et al. gave a survey of trajectory optimization methods in [36]. Trajectory optimization methods can be classified as either direct or indirect. Both have well-known advantages and disadvantages. Direct methods do not require as much mathematical sophistication (regarding the use thereof) nor as much up-front analysis as indirect methods, but they are also not as efficient as indirect methods. When indirect methods do converge, they usually do so quite rapidly relative to direct methods, especially when the initial guess is good. In a guidance application, initial guesses are typically quite good because each guidance cycle has a solution available from the previous cycle. With indirect methods, there is a lot of sensitivity to initial guesses. A principal objective of this research is to mitigate this sensitivity. Because the guidance function is extremely time-critical, efficiency is an overriding concern. Therefore, the indirect approach and its attendant mathematical theory from the calculus of variations is chosen. The discussion of particular numerical solution methods is deferred to a later section.

The next few subsections will discuss the necessary conditions from COV that must be satisfied for an optimum trajectory. The next subsection will discuss the costate differential equations, followed by a subsection on the treatment of the maximum principle to obtain the optimum control (thrust direction) in terms of the state and costate, and finally a discussion of transversality conditions.

2.4 Necessary Conditions of Optimality

2.4.1 Costate Differential Equations

For the purpose of deriving the costate differential equations, note that the atmospheric portion of flight occurs over a very small ground track, enabling the use of the flat-Earth approximations:

$$h \cong r_x - r_E \tag{2-10}$$

$$\mathbf{g} \cong \begin{bmatrix} g_0 & 0 & 0 \end{bmatrix}^T \tag{2-11}$$

where r_x is the vehicle's position component along the guidance coordinate frame x-axis, r_E is the radius of the Earth, and g_0 is the gravitational acceleration at the initial time. With these, the state equations become, with explicit state dependencies called out:

$$\begin{aligned}
\dot{r}_x &= v_x \\
\dot{r}_y &= v_y \\
\dot{r}_z &= v_z \\
\dot{v}_x &= -g_0 + \frac{\left[T(r_x) - A(r_x, \mathbf{v})\right] x_{b_x} - N(r_x, \mathbf{v}) z_{b_x}}{m} \\
\dot{v}_y &= \frac{\left[T(r_x) - A(r_x, \mathbf{v})\right] x_{b_y} - N(r_x, \mathbf{v}) z_{b_y}}{m} \\
\dot{v}_z &= \frac{\left[T(r_x) - A(r_x, \mathbf{v})\right] x_{b_z} - N(r_x, \mathbf{v}) z_{b_z}}{m}
\end{aligned} \tag{2-12}$$

Note that the simplified equations of motion (2-12) are used only in the derivation of the costate equations so that the only unwanted side effect is a potential degradation of optimality in problems that grossly violate the flat-Earth approximations. Define the Hamiltonian, H

$$
\begin{aligned}
H &= -\frac{T_{vac}}{c} + \boldsymbol{\lambda}_{\mathbf{r}}^{T}\mathbf{v} + \boldsymbol{\lambda}_{\mathbf{v}}^{T}\dot{\mathbf{v}} \\
&= -\frac{T_{vac}}{c} + \boldsymbol{\lambda}_{\mathbf{r}}^{T}\mathbf{v} + \lambda_{v_x}\left\{-g_0 + \frac{\left[T(r_x) - A(r_x,\mathbf{v})\right]x_{b_x} - N(r_x,\mathbf{v})z_{b_x}}{m}\right\} + \\
&\lambda_{v_y}\left\{\frac{\left[T(r_x) - A(r_x,\mathbf{v})\right]x_{b_y} - N(r_x,\mathbf{v})z_{b_y}}{m}\right\} + \lambda_{v_z}\left\{\frac{\left[T(r_x) - A(r_x,\mathbf{v})\right]x_{b_z} - N(r_x,\mathbf{v})z_{b_z}}{m}\right\}
\end{aligned}
\tag{2-13}
$$

This results in the following simplification (after Vinh [37]) in the *y* and *z* components of the position costate differential equations:

$$
\dot{\lambda}_{r_y} = \dot{\lambda}_{r_z} = 0 \tag{2-14}
$$

The rest of the costate equations are given by:

$$
\begin{aligned}
\dot{\lambda}_{rx} &= -\frac{\partial H}{\partial r_x} = \frac{-(T_h - A_h)\boldsymbol{\lambda}_{\mathbf{v}}^{T}\mathbf{x_b} + N_h\boldsymbol{\lambda}_{\mathbf{v}}^{T}\mathbf{z_b}}{m} \\
\dot{\lambda}_{vx} &= -\frac{\partial H}{\partial v_x} = -\lambda_{r_x} + \frac{A_{v_x}\boldsymbol{\lambda}_{\mathbf{v}}^{T}\mathbf{x_b} + N_{v_x}\boldsymbol{\lambda}_{\mathbf{v}}^{T}\mathbf{z_b}}{m} \\
\dot{\lambda}_{vy} &= -\frac{\partial H}{\partial v_y} = -\lambda_{r_y} + \frac{A_{v_y}\boldsymbol{\lambda}_{\mathbf{v}}^{T}\mathbf{x_b} + N_{v_y}\boldsymbol{\lambda}_{\mathbf{v}}^{T}\mathbf{z_b}}{m} \\
\dot{\lambda}_{vz} &= -\frac{\partial H}{\partial v_z} = -\lambda_{r_z} + \frac{A_{v_z}\boldsymbol{\lambda}_{\mathbf{v}}^{T}\mathbf{x_b} + N_{v_z}\boldsymbol{\lambda}_{\mathbf{v}}^{T}\mathbf{z_b}}{m}
\end{aligned}
\tag{2-15}
$$

where the subscripts *h, v_x, v_y*, and v_z denote partial differentiation with respect to those variables. No known analytic solutions for the atmospheric state/costate system exist so that numerical integration or some other numerical method is required to obtain the atmospheric state/costate solution.

2.4.2 Optimality Condition

Applying the maximum principle to the Hamiltonian results in the optimization sub-problem:

$$\max_{\mathbf{x_b}}\left\{\boldsymbol{\lambda}_{\mathbf{v}}^{T}\left(\frac{(T-A(\mathbf{x_b}))\mathbf{x_b}-N(\mathbf{x_b})\mathbf{z_b}}{m}\right)\right\} \tag{2-16}$$

Note that the optimal control, $\mathbf{x_b}^{\mathbf{o}}$ (and hence, optimal *z*-body axis, $\mathbf{z_b}^{\mathbf{o}}$) lies in the plane defined by the initial position and velocity-costate vectors[2]. Thus, with reference to Figure 2-2 (after Gath and Calise [25]), the optimization sub-problem can be written simply as

$$\max_{\delta}\{(T-A(\delta))\cos(\delta)+N(\delta)\sin(\delta)\} \tag{2-17}$$

Note that angle of attack is being treated as a function of δ; hence, the dependence of axial and normal forces on δ called out in (2-17). It is necessary, then, to be able to evaluate α in terms of δ so that A and N can be evaluated in (2-17). From Figure 2-2, it is clear that α is some constant, α_0, (constant with respect to δ, that is) minus δ:

$$\alpha = \alpha_0 - \delta \tag{2-18}$$

[2] For suppose that $\mathbf{x_b}^{\mathbf{o}}$ has a component that lies outside the $\mathbf{r_0}$-$\boldsymbol{\lambda}\mathbf{v}$ plane. Then $\mathbf{x_b}^{\mathbf{o}}$ can be expressed as the sum of two vectors, one in-plane, the other, out-of-plane. The out-of-plane vector makes no contribution to the Hamiltonian and can be taken as zero. Thus, an in-plane vector alone is sufficient to extremize the Hamiltonian.

The formula for α_0 can be derived by setting δ to zero and using the standard definition for angle of attack. The components of the wind-relative velocity vector along the *x*- and *z*-body axes (whenever $\delta = 0$) are given by:

$$\begin{aligned} v_{x_b} &= \mathbf{v}_r^T \mathbf{x_b} = v_r \cos(\phi_0) \\ v_{z_b} &= \mathbf{v}_r^T \mathbf{z_b} = \pm \mathbf{v}_r^T \frac{\mathbf{x_b} \times (\mathbf{x_b} \times \hat{\mathbf{r}}_\mathbf{0})}{|\mathbf{x_b} \times \hat{\mathbf{r}}_\mathbf{0}|} = \pm \mathbf{v}_r^T \frac{(\mathbf{x_b}^T \hat{\mathbf{r}}_\mathbf{0})\mathbf{x_b} - \hat{\mathbf{r}}_\mathbf{0}}{\sin(\phi)} \\ &= \pm \mathbf{v}_r^T \frac{\cos(\phi)\mathbf{x_b} - \hat{\mathbf{r}}_\mathbf{0}}{\sin(\phi)} = \pm \frac{\left[v_r \cos(\phi)\cos(\phi_0) - v_{r_x}\right]}{\sin(\phi)} \end{aligned} \tag{2-19}$$

With these, the following results:

$$\alpha_0 = \tan^{-1}\left(\frac{v_{z_b}}{v_{x_b}}\right) = \tan^{-1}\left\{\frac{\pm\left[v_r \cos(\phi)\cos(\phi_0) - v_{r_x}\right]}{v_r \sin(\phi)\cos(\phi_0)}\right\} \tag{2-20}$$

In (2-20), note that ϕ and ϕ_0, and hence, α_0, are simple functions of the state and costate.

The maximization sub-problem (2-17) can be solved in many ways. One option is to take the derivative with respect to δ, set to zero and use an iterative procedure (e.g., Newton's method) to get the root which corresponds to the optimum δ. This approach was found to be problematic because there are situations when the Hamiltonian (as a function of δ) is very flat and Newton's method is very slow to converge. A more direct method is to do a Golden Section search [38]. The Golden Section method, as far as minimization methods go, is inefficient (in terms of number of function evaluations) but the function to be optimized in this case, the Hamiltonian, is fairly inexpensive to evaluate. Brent's 1-dimensional optimization algorithm [38] is more efficient than golden search but perhaps not enough, in this application, to justify the more complicated code.

Figure shows the control-dependent portion of the Hamiltonian as a function of δ at a point in the trajectory very near maximum dynamic pressure. This clearly shows that the sufficiency condition, $H_{\delta\delta} > 0$, is satisfied. For the case of zero or low dynamic pressure, the control-dependent part of the Hamiltonian reduces to $T\cos(\delta)$ which always satisfies the second variation test.

Regardless of how the optimal value of delta, δ^o, is determined, δ^o satisfies the equation

$$\frac{\partial H}{\partial \delta} = -T\sin\left(\delta^o\right) - A_\delta \cos\left(\delta^o\right) + A\sin\left(\delta^o\right) + N_\delta \sin\left(\delta^o\right) + N\cos\left(\delta^o\right) = 0 \tag{2-21}$$

Equation (2-21) is needed in the variational equations as described in Appendix D.

Once δ_0 is determined, the x- and z-body axes are constructed so that the right-hand sides of the state/costate differential equations can be evaluated. Start by expressing the x-body axis, $\mathbf{x_b}$, as a linear combination of costate and initial position vector

$$\mathbf{x_b} = a_1\hat{\boldsymbol{\lambda}}_\mathbf{v} + b_1\hat{\mathbf{r}}_\mathbf{0} \tag{2-22}$$

Dot the preceding with the velocity costate and then with initial position to get

$$\begin{aligned} \cos(\delta) &= \hat{\boldsymbol{\lambda}}_\mathbf{v} \cdot \mathbf{x_b} = a_1 + b_1\cos(\phi) \\ \cos(\phi+\delta) &= \hat{\mathbf{r}}_\mathbf{0} \cdot \mathbf{x_b} = a_1\cos(\phi) + b_1 \end{aligned} \tag{2-23}$$

Solve the preceding two equations for a_1 and b_1 in terms of the angles ϕ and δ:

$$a_1 = \frac{\cos(\delta) - \cos(\phi)\cos(\phi+\delta)}{\sin^2(\phi)}$$
$$b_1 = \frac{\cos(\phi+\delta) - \cos(\phi)\cos(\delta)}{\sin^2(\phi)} \tag{2-24}$$

These expressions are singular when ϕ equals zero. When this is the case, the entire premise for maximizing the Hamiltonian over a single angle (δ) instead of maximizing over the *x*-body axis breaks down. This situation corresponds to zero-valued horizontal components of the primer vector which means that the horizontal components of the velocity vector can be perturbed without affecting the cost. If the mission objectives include going to a target inclination, then perturbing the out-of-plane component of the velocity vector will certainly affect the cost. Therefore, when inclination is constrained, $\phi \neq 0$. This covers most practical cases because it is very typical that attaining a specified inclination is one of the mission objectives. Nevertheless, computer code should be designed to protect against this degenerate case by limiting ϕ to $\pm\varepsilon$ where ε is a small positive number.

Now the *z*-body axis is constructed via:

$$\mathbf{z_b} = \frac{\pm 1}{\sin(\phi+\delta)}\mathbf{x_b}\times\mathbf{y_b} = \frac{\pm 1}{\sin(\phi+\delta)}\mathbf{x_b}\times(\mathbf{x_b}\times\hat{\mathbf{r}}_\mathbf{0}) = \frac{\pm 1}{\sin(\phi+\delta)}\left[(\mathbf{x_b}\cdot\hat{\mathbf{r}}_\mathbf{0})\mathbf{x_b} - \hat{\mathbf{r}}_\mathbf{0}\right]$$
$$= \frac{\pm 1}{\sin(\phi+\delta)}\begin{bmatrix} (a_1\cos(\phi)+b_1)^2 - 1 \\ a_1\lambda_{v_y}(a_1\cos(\phi)+b_1) \\ a_1\lambda_{v_z}(a_1\cos(\phi)+b_1) \end{bmatrix} \tag{2-25}$$

where the fact that the initial position vector has zero *y*- and *z*-components when expressed in the guidance frame is used. Note that this expression is singular when $\phi + \delta = 0$. This situation corresponds to vertical flight so this is potentially problematic early in the ascent. To protect against this, desired *x*- and *z*-body directions could be stored and used in place of the above

expressions whenever $\phi + \delta$ is very near zero. No precautions of this kind were found to be necessary in the current research but if the algorithm undergoes testing in a high-fidelity six-degree-of-freedom simulation, this could show up as a problem that would need to be mitigated as suggested.

The path constraints are treated in the same manner as in Gath and Calise [25]. If the angle of attack resulting from the solution of (2-17) is such that the maximum $q\alpha$ constraint is violated, then α is given by:

$$\alpha_L = \frac{(q\alpha)_{\max}}{q} \tag{2-26}$$

A constraint multiplier, ε, is then calculated from:

$$\left\{ \frac{|\boldsymbol{\lambda}_{\mathbf{v}}|}{m} \left[(T - A + N_\delta)\sin(\delta) + (A_\delta - N)\cos(\delta) \right] + \varepsilon q \right\}_{\alpha=\alpha_L} = 0 \tag{2-27}$$

The Euler-Lagrange differential equations are then modified according to:

$$\begin{aligned} \dot{\boldsymbol{\lambda}}_{\mathbf{r}} &= -H_{\mathbf{r}} - \varepsilon \frac{\partial(q\alpha)}{\partial \mathbf{r}} \\ \dot{\boldsymbol{\lambda}}_{\mathbf{v}} &= -H_{\mathbf{v}} - \varepsilon \frac{\partial(q\alpha)}{\partial \mathbf{v}} \end{aligned} \tag{2-28}$$

In the next subsection, the transversality conditions, which impose boundary conditions at the end-point are discussed.

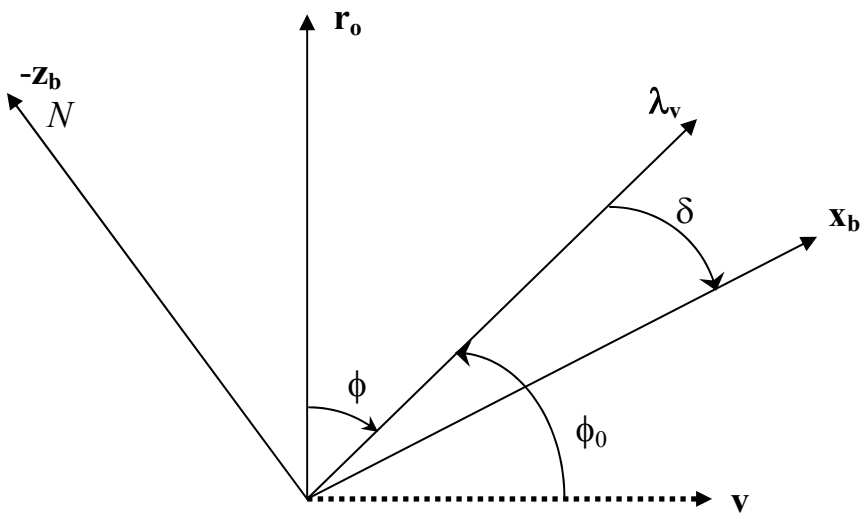

Figure 2-2: Angles Pertaining To Optimality Condition

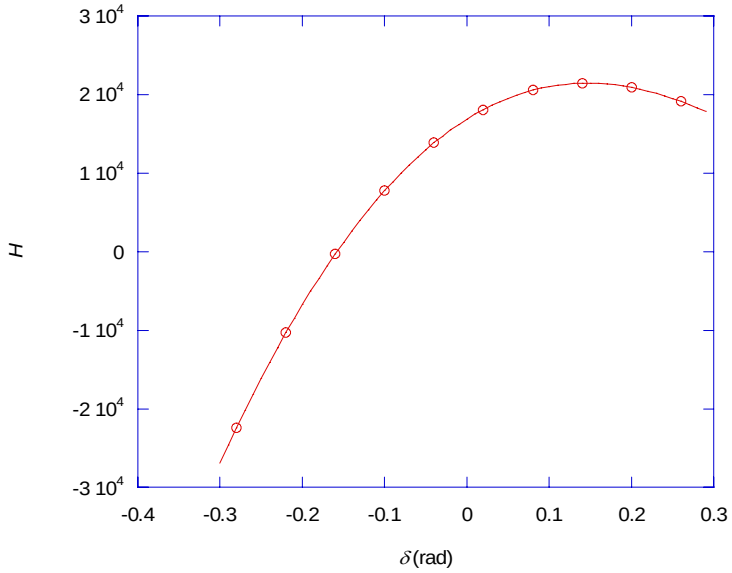

Figure 2-3: Second Variation Test: Hamiltonian vs. δ

2.4.3 Transversality Conditions

The transversality conditions provide end-point constraints on the costates and the Hamiltonian [7]:

$$\begin{aligned} \boldsymbol{\lambda}^{\mathrm{T}}\left(t_f\right) &= \boldsymbol{\nu}^{\mathrm{T}}\boldsymbol{\psi}_{\mathbf{x}}\left(\mathbf{x}\left(t_f\right)\right) \\ H\left(t_f\right) &= 0 \end{aligned} \tag{2-29}$$

where $\boldsymbol{\nu}$ is a vector of constant Lagrange multipliers and $\boldsymbol{\psi}$ is the vector of terminal state constraints. The first of (2-29) are discussed in detail in Chapter 5.

Note that the constraint on $H(t_f)$ is trivial to satisfy because the costate differential equations are linear in costate and homogeneous. Thus, the costates can be scaled by any positive factor, so that the transversality conditions are all satisfied, without affecting the trajectory or any of the necessary conditions [14]. In lieu of pre-selecting an arbitrary component of the initial costate (hence removing that component as one of the parameters to be iteratively determined), the constraint $| \boldsymbol{\lambda}_{\mathbf{v}}(t_f) | = 1$ is added. This allows the initial velocity vector costate to always be normalized to unity, $| \boldsymbol{\lambda}_{\mathbf{v}}(t_0) | = 1$. To sum up, there are 7 end-point boundary conditions (the k terminal state constraints plus the 6-k transversality conditions plus the non-constraining constraint $| \boldsymbol{\lambda}_{\mathbf{v}}(t_f) | = 1$) and 7 parameters (6 components of initial costate plus the time of flight) for which to solve iteratively.

Next candidate numerical methods for solving the resulting two-point boundary-value problem are discussed.

2.5 Numerical Solution Methods

Here, the myriad of solution methods for solving the two-point boundary-value problem formulated above are considered. Keller [39], Stoer and Bulirsch [40], and Press, et al. [38] discuss solution methods, including quasilinearization, finite difference methods, ordinary shooting and multiple shooting. In [25], a modified collocation method employing homotopy on the aerodynamic terms is used to solve the two-point boundary-value problem. In [41], a finite-difference approach is used. Among the simplest and easiest methods to implement is the shooting method which consists of guessing missing initial values, numerically integrating the state/costate system to the predicted cutoff time, constructing the terminal constraints error vector and associated Jacobian matrix and applying Newton's method to generate corrections to the guessed initial values. The finite difference method involves setting up a mesh of (typically equally-spaced) nodes and adjusting states/costates guesses at those nodes until the state/costate differential equations and boundary conditions are satisfied. Various discretization methods can be used with high-order methods to reduce the number of necessary nodes (and guessed values) compared to low-order methods. Multiple shooting is a variant of ordinary shooting wherein intermediate shooting points are introduced, allowing state/costate guesses at these points. This introduces discontinuities in the state/costate histories but these are taken out as the solution progresses by enforcing ancillary continuity constraints. The result is that ill-conditioning is dramatically reduced, even with the introduction of just one intermediate shooting point. Problems that are just a little too ill-conditioned to solve via ordinary shooting, can often be solved quite easily with the introduction of at most a handful of intermediate shooting points. Thus, the Jacobian matrix and associated linear system at the terminal point stays relatively small. Like the finite differences method, the associated Jacobian matrix is very sparse and thus can be solved efficiently with a sparse method rather than, say, straightforward Gauss elimination [40].

Unlike finite difference methods, the solution accuracy (and required accuracy of the numerical integrator) of the multiple shooting method is independent of the number of mesh/shooting points, resulting in a more modular algorithm. The research discussed in the sequel uses the multiple shooting method for solving the two-point boundary-value problem.

A brief summary of the solution process is as follows.

1. Guess values for the time of flight, missing costates at the initial (current 'mission elapsed') time and states/costates at intermediate time points. These values come from the previous guidance cycle or, if no previous values exist, from a pre-specified straight-line approximation to the states/costates or perhaps from an in-vacuo solution.
2. At each shooting point, the state/costate differential equations are initialized with the guessed values from step 1, then numerically (or analytically, if 'outside' the atmosphere) integrated (using, e.g., Runge-Kutta) to the next shooting point. These integrations can be done in parallel if desired. Store the state/costate solutions at each intermediate shooting point and at the final boundary.
3. With the solutions from step 2, evaluate the continuity conditions at the shooting points, the *k* terminal state constraints, and the *6-k* transversality conditions.
4. Repeat steps 2 and 3, as required, to obtain Jacobian information. Jacobian information can be obtained using finite differencing or variational equations or a combination thereof. Several commercial off-the-shelf software packages are currently available as an aid in developing partial derivatives of (e.g., FORTRAN or C code) algebraic expressions if needed. The choice between the finite differencing approach and variational equations approach is largely a matter of personal preference as there are pros and cons to each method. The main disadvantage of the

variational approach to obtaining derivatives is that it significantly increases the analytical and coding burden. The approach used throughout this research is the variational approach, resulting in derivatives accuracy to within machine accuracy. Appendix D describes in detail the variational approach.

5. Compute the Newton correction step/direction required to null all (linearized) constraints of step 3. This involves (e.g., Gauss elimination) solution of a 6×6 (fixed final time) or 7×7 (free final time) linear system to get the initial costate and time of flight corrections followed by a back solving process [40] to get the corrections to be applied to the state/costate guesses at the intermediate shooting points.
6. Limit the correction step so that the applied corrections are not too large. Experimentation shows that an approximate 25 percent change limit on the initial costates and time of flight is sufficient to provide reliable convergence.
7. Go to step 2 with the corrected guesses for states/costates and time of flight. Repeat until, e.g., the sum of the squares of all constraint violations is less than a specified tolerance.

Chapter 4 presents guided trajectory results using the atmospheric formulation presented here and the vacuum formulation of the next chapter.

CHAPTER 3

VACUUM GUIDANCE FORMULATION

Most of ascending flight occurs at low dynamic pressure such that atmospheric effects can be neglected, thus a separate vacuum guidance formulation is useful. It is shown in this chapter that closed-form solutions for the states and costates can be obtained under the assumption of uniform gravity and linear steering. These assumptions can be relaxed to the extent desired simply by dividing the trajectory into multiple segments and re-applying the closed-form solution. Ascent trajectories are characterized by small spatial variation. Hence, the assumptions are typically excellent, with the results consisting of guided trajectories that are indistinguishable from fully optimized trajectories. For guided applications, as the time-to-go becomes small, the approximation quality improves, which in turn reduces any deleterious effects caused by otherwise gross approximations. Although the particular application here is ascent guidance, the formulation is equally suitable for orbital transfer guidance.

Burrows and McDaniel [10] treat the gravity integrals using high-order Taylor series expansions. McHenry et al. [3] use numerical integration to obtain the gravity integrals and Calise et al. [24] use numerical quadrature to treat the thrust integrals. The solution presented here is free of such numerical treatments. The equations of motion for a rocket under the influence of thrust, T, and gravity, $\mathbf{g(r)}$, are

$$\begin{aligned} \dot{\mathbf{r}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= \mathbf{g}(\mathbf{r}) + \frac{T}{m}\hat{\mathbf{u}}(t) \end{aligned} \tag{3-1}$$

Assume a linear thrust steering law of the form

$$\hat{\mathbf{u}}(t) = \frac{\mathbf{A}t + \mathbf{B}}{|\mathbf{A}t + \mathbf{B}|} \tag{3-2}$$

where **A** and **B** are constant 3-vectors. This steering form is justified later in this chapter. The first and second thrust integrals for a constant thrust flight phase, of duration *t*, are given by [10,42]

$$\begin{aligned}
\Delta\mathbf{v}_T &= \frac{T}{\dot{m}}\left[\frac{1}{A}\ln\left(\frac{e}{g}\right) + \frac{m_0}{d}\ln\left(\frac{p}{q}\right)\right]\mathbf{A} + \frac{T}{d}\ln\left(\frac{p}{q}\right)\mathbf{B} \\
\Delta\mathbf{r}_T &= f_{11}T\mathbf{A} + T\left[-\frac{1}{A\dot{m}}\ln\left(\frac{e}{g}\right) - \frac{m(t)}{d\dot{m}}\ln\left(\frac{p}{q}\right)\right]\mathbf{B}
\end{aligned} \tag{3-3}$$

where the auxiliary quantities in (3-3) are

$$\begin{aligned}
d &= \sqrt{(m_0\mathbf{A} + \dot{m}\mathbf{B})^T(m_0\mathbf{A} + \dot{m}\mathbf{B})} \\
e &= \mathbf{A}^T\mathbf{B} + AB \\
f &= \sqrt{(\mathbf{A}t + \mathbf{B})^T(\mathbf{A}t + \mathbf{B})} \\
g &= \mathbf{A}^T\mathbf{B} + A^2t + fA \\
p &= m_0\left(B^2\dot{m} + A^2m_0t + \mathbf{A}^T\mathbf{B}(m_0 + \dot{m}t) + df\right) \\
q &= m(t)\left(dB + m_0\mathbf{A}^T\mathbf{B} + \dot{m}B^2\right)
\end{aligned} \tag{3-4}$$

$$\tau = \frac{m_0}{\dot{m}}$$
$$A = |\mathbf{A}|$$
$$B = |\mathbf{B}|$$
$$\dot{m} = \frac{T}{c}$$
$$m_0 = m(t = 0)$$
$$m(t) = m_0 - \dot{m}t$$
$$f_{11} = \frac{T}{\dot{m}}\left[\frac{f - B}{A^2} + \ln\left(\frac{e}{g}\right)\frac{\left(t - \tau + \frac{A^T B}{A^2}\right)}{A} + \ln\left(\frac{p}{q}\right)m_0\frac{(t - \tau)}{d}\right]$$

Another common flight phase for which thrust integrals are needed is one in which the acceleration due to thrust is constant. An example is when the axial acceleration magnitude is constrained. In such a case, the vehicle reduces its throttle to maintain a constant acceleration. It turns out that the thrust integrals for constant thrust can be used by setting the rate of change of mass to a small positive value proportional to the initial mass, m_0, divided by the phase duration t:

$$\dot{m} = t\frac{m_0}{1000} \tag{3-5}$$

and using this value in the evaluation of the thrust integrals. This is necessary because the thrust integrals are undefined when the mass rate is zero. An alternative of course would be to obtain and code the thrust integral expressions for the case of constant thrust acceleration and modify the guidance logic so that thrust integrals appropriate to the flight phase are used.

The position and velocity increments, (3-3), are due to the action of thrust alone. To incorporate the effects of gravity, an average gravity approach is used. The "predicted" position vector at the time t is calculated assuming a constant gravity field and using the thrust integral

$$\mathbf{r}_p(t) = \mathbf{r}_0 + \mathbf{v}_0 t + \Delta\mathbf{r}_T - \frac{t^2}{2}\frac{\mu}{|\mathbf{r}_0|^3}\mathbf{r}_0 \tag{3-6}$$

The "corrected" position and velocity vectors in vacuum flight at time t are given by

$$\begin{aligned}\mathbf{r}(t) &= \mathbf{r}_0 + \mathbf{v}_0 t + \Delta\mathbf{r}_T + \frac{t^2}{2}\mathbf{g}_{ave} \\ \mathbf{v}(t) &= \mathbf{v}_0 + \Delta\mathbf{v}_T + t\mathbf{g}_{ave}\end{aligned} \tag{3-7}$$

where the average gravity, $\mathbf{g}_{ave}$, is the average of the initial gravity vector and the gravity vector at the predicted position

$$\mathbf{g}_{ave} = -\frac{\mu}{2}\left(\frac{1}{|\mathbf{r}_0|^3} + \frac{1}{|\mathbf{r}_p|^3}\right) \tag{3-8}$$

The approximation of optimal steering by the linear steering law (3-2) is suggested by considering the problem of in-vacuo flight in a uniform gravity field. The Hamiltonian is given by

$$H = \boldsymbol{\lambda}_r^T \mathbf{v} + \boldsymbol{\lambda}_v^T\left(\mathbf{g} + \frac{T}{m}\hat{\mathbf{u}}\right) \tag{3-9}$$

where **g** is a constant gravity acceleration vector, which can be considered to be an average or effective gravity vector. The costates are given by

$$\begin{aligned}\dot{\boldsymbol{\lambda}}_{\mathbf{r}}^{T} &= -H_{\mathbf{r}} = 0 \Rightarrow \boldsymbol{\lambda}_{\mathbf{r}}(t) = -\mathbf{A} \\ \dot{\boldsymbol{\lambda}}_{\mathbf{v}}^{T} &= -H_{\mathbf{v}} = -\boldsymbol{\lambda}_{\mathbf{r}}^{T} = \mathbf{A}^{T} \Rightarrow \boldsymbol{\lambda}_{\mathbf{v}}(t) = \mathbf{A}t + \mathbf{B}\end{aligned} \tag{3-10}$$

The maximum principle implies that the optimal thrust direction is along the primer vector, $\boldsymbol{\lambda}_{\mathbf{v}}$

$$\hat{u}(t) = \frac{\boldsymbol{\lambda}_{\mathbf{v}}(t)}{\left|\boldsymbol{\lambda}_{\mathbf{v}}(t)\right|} = \frac{\mathbf{A}t + \mathbf{B}}{\left|\mathbf{A}t + \mathbf{B}\right|} \tag{3-11}$$

Comparing (3-10) and (3-11), it is apparent that the **A** and **B** vectors should be set according to

$$\begin{aligned}\mathbf{A} &= -\boldsymbol{\lambda}_{\mathbf{r}}(t_0) \\ \mathbf{B} &= \boldsymbol{\lambda}_{\mathbf{v}}(t_0)\end{aligned} \tag{3-12}$$

where the initial time, t_0, in (3-12), denotes the time at the beginning of the propagation interval.

It is interesting to note some properties of the linear steering law (3-2). The thrust direction rotates about a fixed axis normal to the plane containing the **A** and **B** vectors at a time-varying rotational (attitude maneuver) rate given by

$$\omega(t) = \frac{\left|\mathbf{u} \times \dot{\mathbf{u}}\right|}{\left|\mathbf{u}\right|^{2}} = \frac{\left|\mathbf{A} \times \mathbf{B}\right|}{A^{2}t^{2} + 2\mathbf{A}^{T}\mathbf{B}t + B^{2}} \tag{3-13}$$

At time zero, the time of the guidance update, the attitude maneuver rate is given by

$$\omega(t=0)=\frac{\left|\mathbf{A}\times\mathbf{B}\right|}{\mathbf{B}^T\mathbf{B}}=\frac{A\sin(\angle\mathbf{AB})}{B} \tag{3-14}$$

The time at which the maximum maneuver rate occurs is determined by minimizing the denominator in (3-13) to get:

$$t_{\omega_{\max}}=-\frac{\mathbf{A}^T\mathbf{B}}{\mathbf{A}^T\mathbf{A}}=-\frac{B\cos(\angle\mathbf{AB})}{A} \tag{3-15}$$

The time of maximum angular rate is limited to lie within the interval [t_0, t_f] where t_0 is the time of the guidance update and t_f is the predicted cutoff time. The time is then substituted into the general turning rate expression to get the maximum maneuver rate for the burn. This parameter can be used as an indicator of infeasibility after the calculation of candidate abort trajectories. For example, for a large thrust loss occurring near nominal MECO, the new guidance solution may require large maneuver rates. The maximum maneuver rate could be computed and checked to ensure it doesn't exceed a maximum allowed maneuver rate. If the anticipated maneuver rate exceeds the maximum allowed, then different cutoff targets/conditions (e.g., different perigee and apogee) can be used and a new guidance solution computed in hopes of obtaining a more benign trajectory.

The equivalence between the classical bi-linear tangent steering law and our steering form can be seen if an auxiliary set of orthogonal *x*-*y* axes defined in the plane defined by the **A** and **B** vectors is visualized. Then, the thrust direction angle with respect to the *x*-axis is given by:

$$\tan(\theta) = \frac{B_y + A_y t}{B_x + A_x t} \tag{3-16}$$

Note that here there are only four unknown parameters. However, two unknown parameters are required to orient the plane containing the locus of thrust directions. For two-dimensional problems, the total number of costates does indeed reduce to four which define the pitch history contained in the plane of motion. The successful application of the linear steering law, or variations of it, is widespread and discussed in [3,5,6,7,10,11,23,46,47].

Next, the costate propagation equations are derived. Assume, only for the purpose of obtaining a vacuum costate solution, a linear central gravity field (see Ref. [42,43]), that is:

$$\mathbf{g}(\mathbf{r}) \cong -\frac{\mu}{r_0^3}\mathbf{r} \equiv -\omega_0^2 \mathbf{r} \tag{3-17}$$

where ω_0 is a constant, so that the costate differential equations become:

$$\begin{bmatrix} \dot{\boldsymbol{\lambda}}_{\mathbf{r}} \\ \dot{\boldsymbol{\lambda}}_{\mathbf{v}} \end{bmatrix} = \begin{bmatrix} \mathbf{0_3} & \omega_0^2 \mathbf{I_3} \\ -\mathbf{I_3} & 0_3 \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda}_{\mathbf{r}} \\ \boldsymbol{\lambda}_{\mathbf{v}} \end{bmatrix} \tag{3-18}$$

with a convenient solution suitable for an in-vacuo guidance scheme, namely,

$$\begin{aligned} \boldsymbol{\lambda}_{\mathbf{r}}(t) &= \cos(\omega_0 t)\boldsymbol{\lambda}_{\mathbf{r}}(t_0) + \omega_0 \sin(\omega_0 t)\boldsymbol{\lambda}_{\mathbf{v}}(t_0) \\ \boldsymbol{\lambda}_{\mathbf{v}}(t) &= -\frac{\sin(\omega_0 t)}{\omega_0}\boldsymbol{\lambda}_{\mathbf{r}}(t_0) + \cos(\omega_0 t)\boldsymbol{\lambda}_{\mathbf{v}}(t_0) \end{aligned} \tag{3-19}$$

Whenever dynamic pressure is reasonably small, say, less than 50 psf, the in-vacuo assumption can be used. Typically, for a given mission, a good initial guess for the mission-elapsed time at which dynamic pressure becomes small is available. Up until that time based on the simulation, the atmospheric equations developed earlier are used in the (numerical) propagation of state and costate. At that point, the current propagated state/costate is handed off to the simpler equations [22]. It is important to do this, that is, only use the atmospheric equations when absolutely necessary, because the atmospheric equations are so much more complicated than the analogous in-vacuo equations. Note that it is not critical that the predicted time of atmospheric exit, $t_{Exo,}$ be especially accurate. If need be, the predicted dynamic pressure at t_{Exo} can be monitored and t_{Exo} increased if necessary.

Chapter 4 presents guided trajectory results using the atmospheric formulation of Chapter 3 and the in-vacuo formulation described here.

CHAPTER 4

SINGLE-STAGE-TO-ORBIT VEHICLE

This chapter presents numerical results using the formulations of Chapters 2 and 3. The objective is to compare guided trajectory parameters with the same parameters from POST open-loop optimal trajectories in order to provide confidence in the formulations. Because our formulation is a closed-loop guidance algorithm whereas POST is an off-line trajectory optimizer, this is not an apples-to-apples comparison. The vehicle and environment models used in the closed-loop guidance algorithm generally do not match the simulation version of the vehicle and environment models whereas the POST optimizer has perfect knowledge of all models. Moreover, the steering form assumed in POST is inherently different than that assumed in the closed-loop guidance. POST is a nonlinear programming code which approximates optimal control problems with a finite set of optimizeable parameters. Despite these inherent differences, the fuel performance and state histories are expected to be comparable. For example, fuel usage is expected to match to within about three hundred pounds out of a total fuel usage of over 200,000 pounds.

The algorithm formulation has been coded in C and implemented as a 1 Hz guidance function into Marshall Space Flight Center's high-fidelity MAVERIC trajectory simulator. A second-order Runge-Kutta numerical integrator is used with 20 integration steps for the propagation of the state/costate atmospheric equations. The end of the atmospheric phase is taken as 140 seconds after liftoff inside the guidance solution process. In each simulation, after the initial guidance cycle, one full Newton iteration is performed to correct the initial costates and time of

flight and the result taken as the solution. The optimized trajectory from the previous guidance cycle is used as the initial guess for the current cycle. One intermediate shooting point is used. It was found that this gives significantly better convergence properties than ordinary shooting[3]. A $q\alpha$ constraint of 500 psf-deg is imposed inside the guidance solution process. For two simulated orbital missions, an axial acceleration limit of 3.5 g's is imposed. The 3 degree-of-freedom simulations run significantly faster than real-time on a late 1990's era DEC alpha computer/processor. The initial guess is generated from the in-vacuo solution. Results for three mission simulations are shown: 1) A six degree-of-freedom sub-orbital mission simulation launching from Edwards Air Force Base to Michael Army Air Field in Utah, 2) a 3 degree-of-freedom orbital mission launching from Kennedy Space Center directly into a 100 by 100 nautical mile (perigee by apogee), 28.5 degree inclination orbit, and 3) a three degree-of-freedom orbital mission launching from Kennedy Space Center to a 40 by 150 nautical mile, 51.6 degree inclination space station rendezvous orbit. For the first mission, the X-33 vehicle model [4] is used and for the remaining two, the X-33 vehicle with increased Isp (to make the vehicle orbit-capable) is used. Appendix E lists X-33 vehicle data. The guidance in all cases is initiated at liftoff and executed at 1 Hz. The guidance scheme is implemented by re-optimizing the remaining trajectory. During the first four seconds after liftoff, the trajectory solution is ignored in the guidance command generation logic. Instead, the guidance commands are set to execute a vertical rise off of the launch pad. In all cases, open-loop reference trajectories (from POST)[4] are available and comparison plots were made to verify acceptable performance of the guidance. In the orbital cases, the guided trajectories are very comparable to the open-loop trajectories with the guided trajectories out-performing the POST trajectories by about 200 pounds of fuel out of a total fuel usage of over 200,000 pounds.

[3] In some flight conditions, ordinary shooting flails around for about one hundred iterations (from a very good initial guess) before converging to a solution.

[4] Grateful acknowledgement to Terri Schmitt, Vehicle Flight Mechanics Group, MSFC.

Figure 4-1 shows the Hamiltonian vs. time over a representative optimized trajectory. The Hamiltonian should be a constant over an optimized trajectory. This fact is traditionally used to provide a numerical consistency check among the state differential equations, the Euler-Lagrange differential equations, and the optimality condition. In the present situation where various approximations to the state equations are invoked before deriving the Euler-Lagrange differential equations, the utility of this check is questionable. The Hamiltonian is fairly close to zero throughout the trajectory but always negative. This is most likely due to the flat-Earth and linear-gravity approximations used in the atmospheric and in-vacuo equations.

Figures 4-2 through 4-8 show sub-orbital trajectory results. Figure 4-2 shows the sub-orbital ascent altitude histories from both the MAVERIC closed-loop and POST open-loop simulations. The curves are noticeably different in the middle of the trajectory due to the different assumed control histories. The final altitudes match to within 20 ft indicating that the guidance is adept at attaining a specified target, which in this case is 175,000 ft. Figure 4-3 shows that the dynamic pressure curves each peak at about 320 psf. Figure 4-4 shows structural load indicators $q\alpha$ and $q\beta$. The guided vehicle is subjected to higher loads due to turbulence in the Global Reference Atmospheric Model (GRAM) wind model. The angles of attack and sideslip are shown in Figure 4-5. Again, due to differences in the winds and the control histories, the aerodynamic angles noticeably differ. The ground track is shown in Figure 4-6. The final segments of the ground track vary because of a subtle difference in lateral targeting. The guidance logic is such that the vehicle's final velocity vector points to the landing site, whereas POST targets for a particular latitude, longitude and velocity heading angle. That is, POST effectively (and unnecessarily) constrains the final down range. The guidance targeting logic is appropriate for this mission where entry guidance takes over steering responsibility immediately after the powered burn. Figure 4-7 shows excellent agreement in the vehicle mass vs. time curves. Figure 4-8 shows the

Euler angles, pitch, yaw, and roll. Apparently, fuel usage is fairly insensitive to variations in attitude history.

Figures 4-9 through 4-15 show plots for the 100 n.m. circular orbit missions. The altitude curves in Figure 4-9 are very similar with differences occurring in the middle due to different attitude histories. The dynamic pressure curves in Figure 4-10 peak at 350 psf. The load indicators and the aerodynamic angles in Figures 4-11 and 4-12 show greater variation in the guided trajectories due to wind turbulence. Figure 4-13 shows a difference in the ground tracks due to the guidance logic targeting for an inclination of 28.5 deg while POST does not constrain inclination. The mass histories in Figure 4-14 are indistinguishable. The pitch angle histories are noticeably different but again this does not affect the performance as the injected weights agree to within 100 lbm.

Figures 4-16 through 4-22 show plots for the 40 by 150 n.m., 51.6 deg inclination mission. The initial portion of the ground tracks in Figure 4-20 varies because the launch pad locations differ by about 5 n.m. The injected masses to orbit differ by just over 200 lbm.

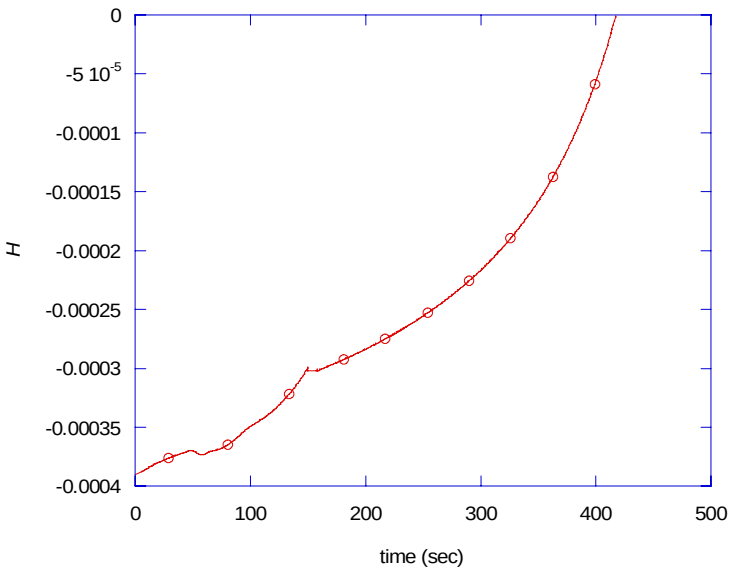

Figure 4-1: Hamiltonian vs. Time

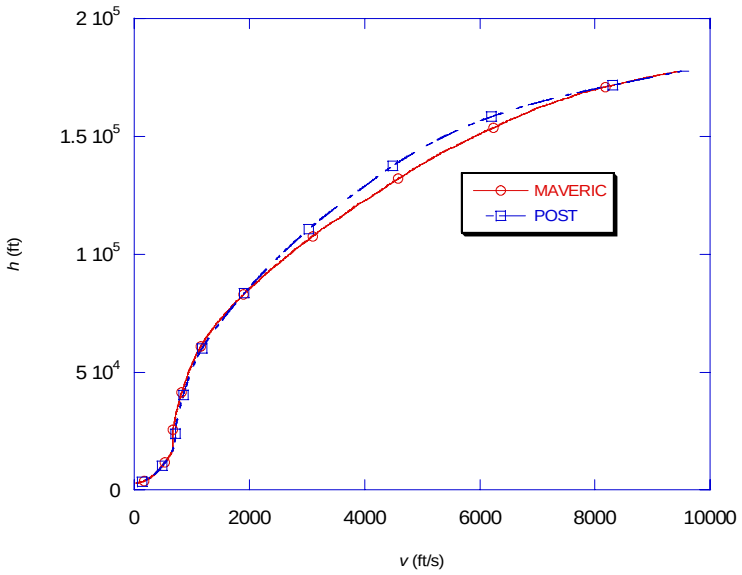

Figure 4-2: Altitude vs Speed, POST/MAVERIC Comparison, Sub-orbital Mission

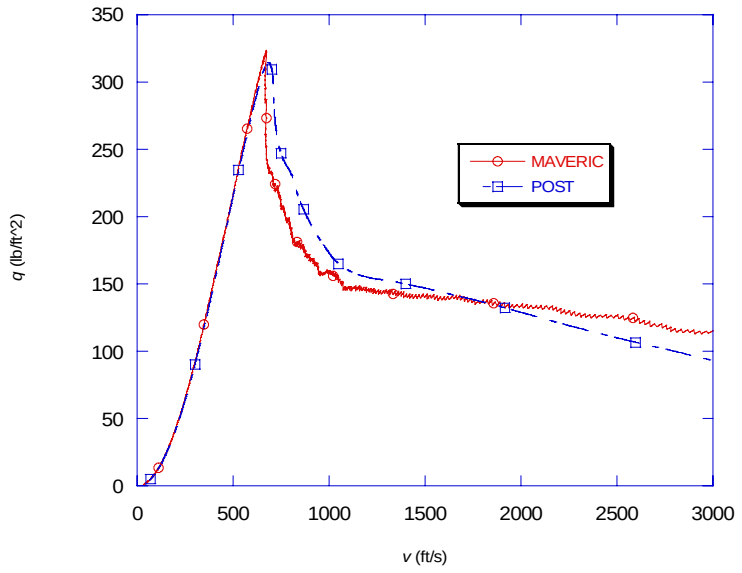

Figure 4-3: Dynamic Pressure vs Speed, POST/MAVERIC Comparison, Sub-orbital Mission

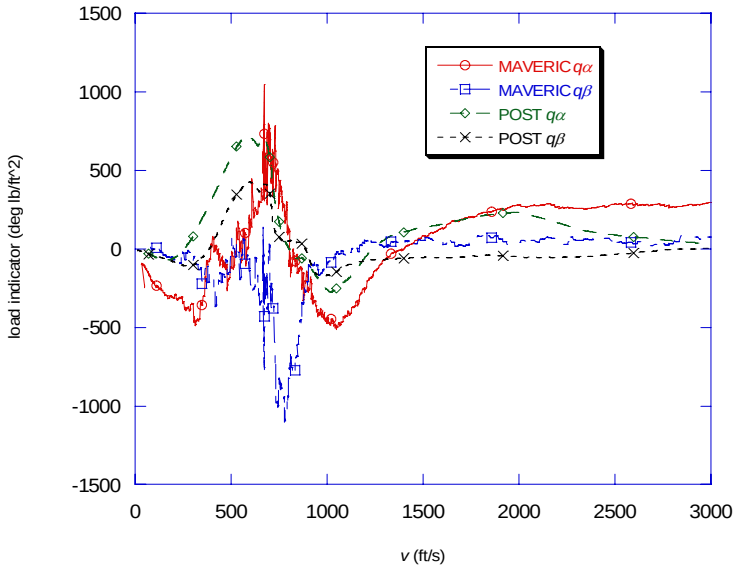

Figure 4-4: $q\alpha$, $q\beta$ vs Speed, POST/MAVERIC Comparison, Sub-orbital Mission

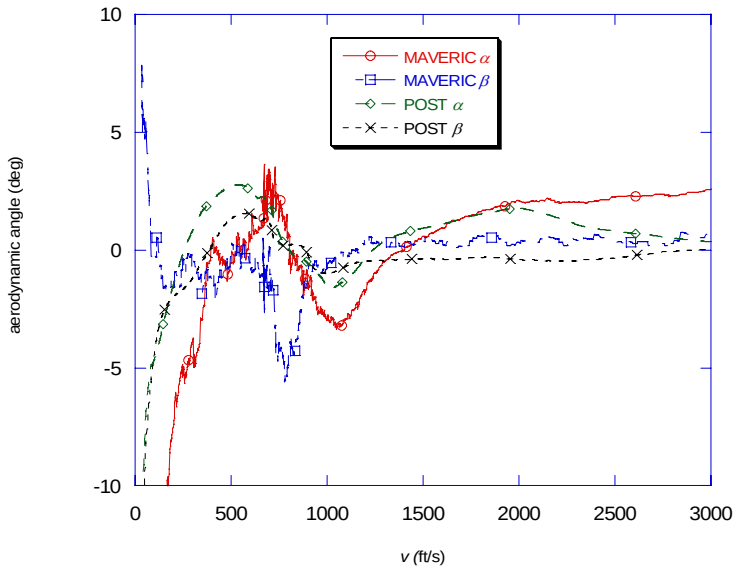

Figure 4-5: α, β vs Speed, POST/MAVERIC Comparison, Sub-orbital Mission

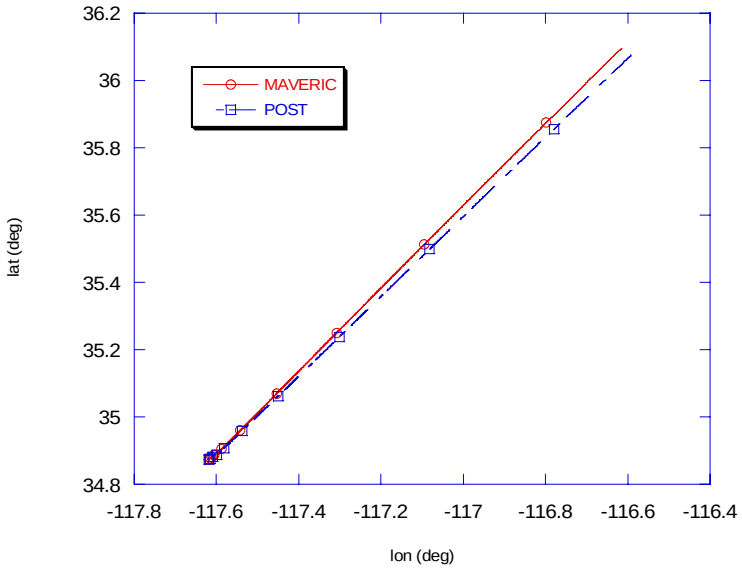

Figure 4-6: Ground Track, POST/MAVERIC Comparison, Sub-orbital Mission

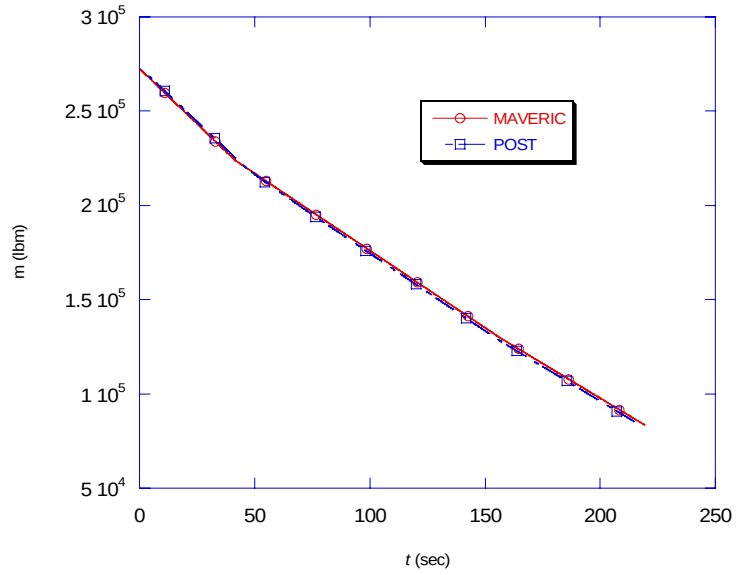

Figure 4-7: Mass vs Time, POST/MAVERIC Comparison, Sub-orbital Mission

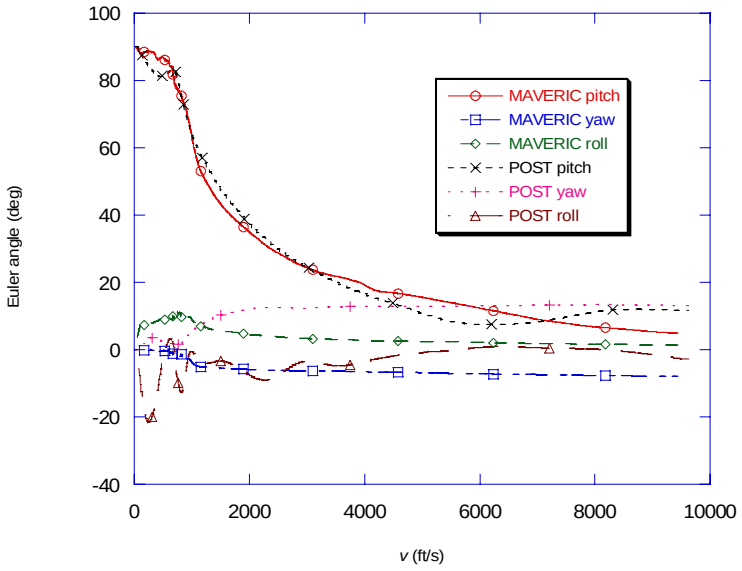

Figure 4-8: Euler Angles vs Speed, POST/MAVERIC Comparison, Sub-orbital Mission

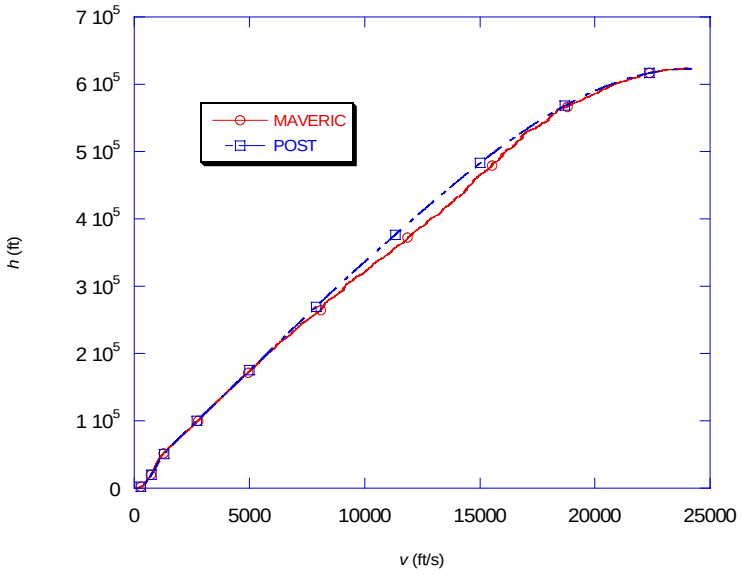

Figure 4-9: Altitude vs Speed, POST/MAVERIC Comparison, Circular Insertion

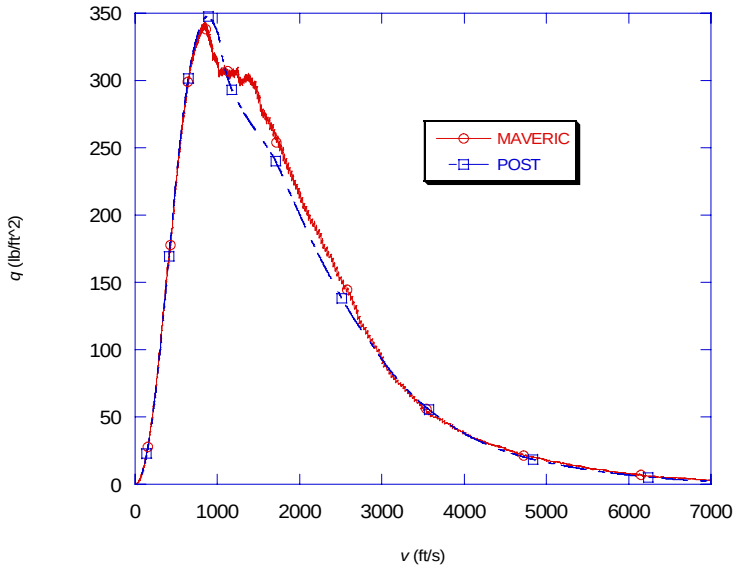

Figure 4-10: Dynamic Pressure vs Speed, POST/MAVERIC Comparison, Circular Insertion

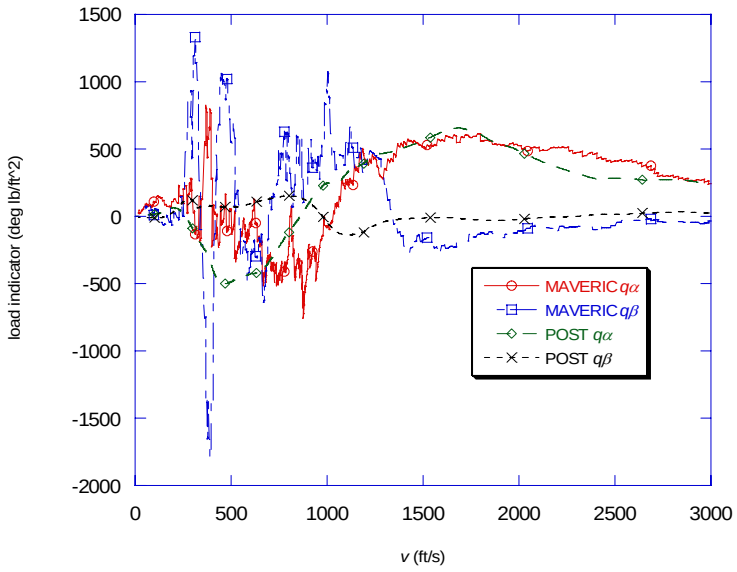

Figure 4-11: $q\alpha$, $q\beta$ vs Speed, POST/MAVERIC Comparison, Circular Insertion

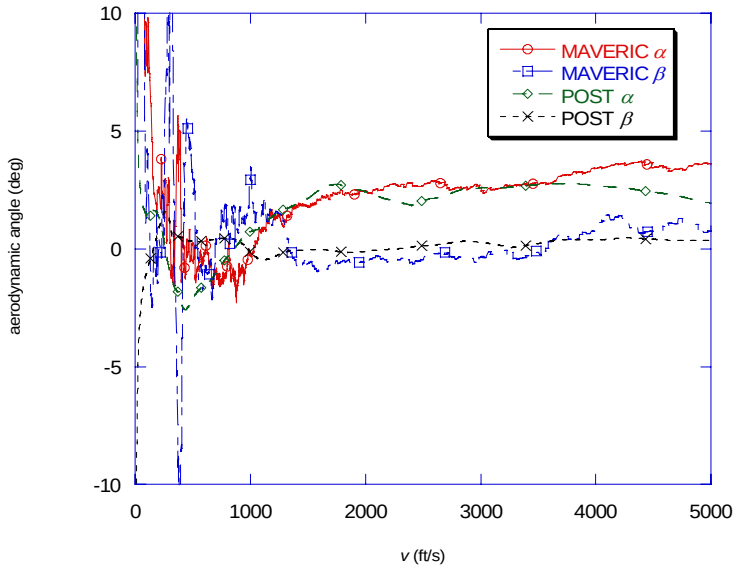

Figure 4-12: α, β vs Speed, POST/MAVERIC Comparison, Circular Insertion

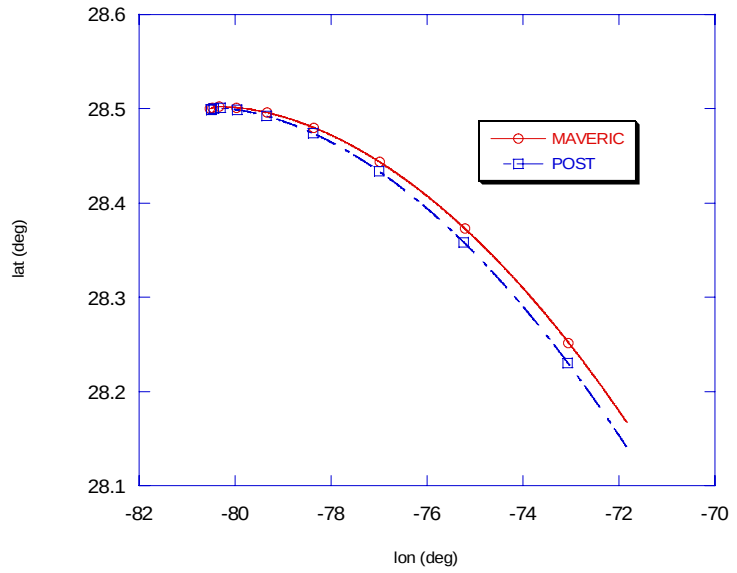

Figure 4-13: Ground Track, POST/MAVERIC Comparison, Circular Insertion

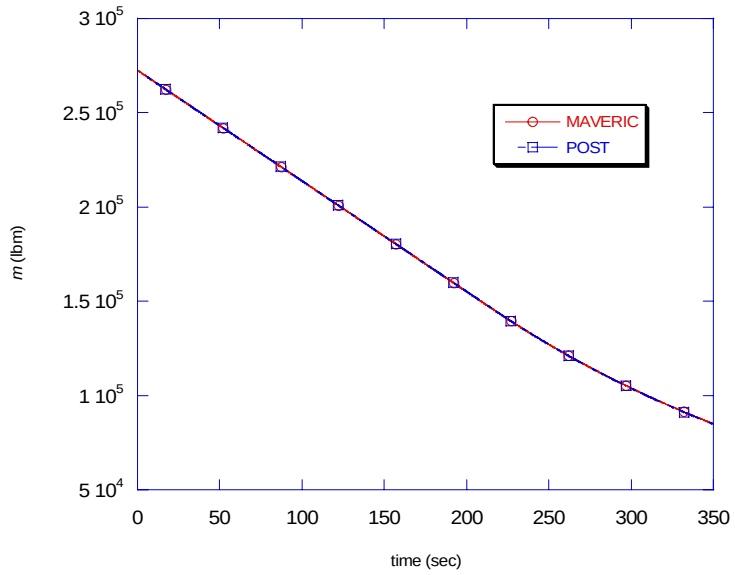

Figure 4-14: Vehicle Mass, POST/MAVERIC Comparison, Circular Insertion

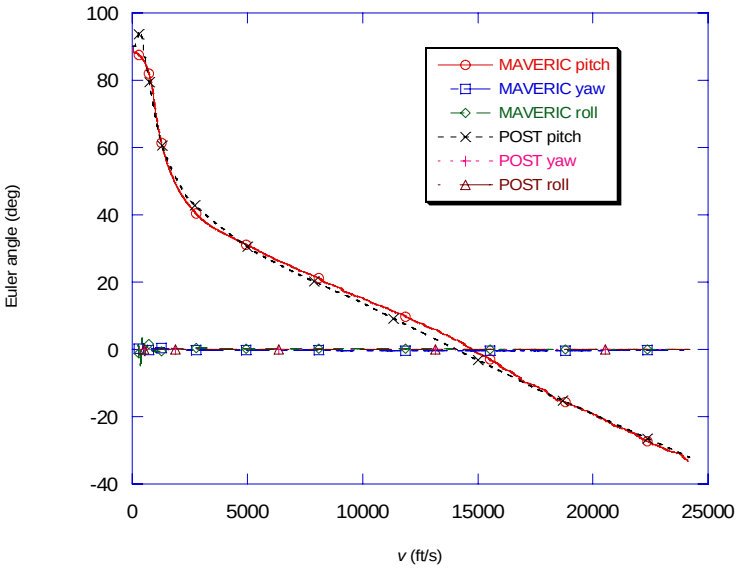

Figure 4-15: Euler Angles, POST/MAVERIC Comparison, Circular Insertion

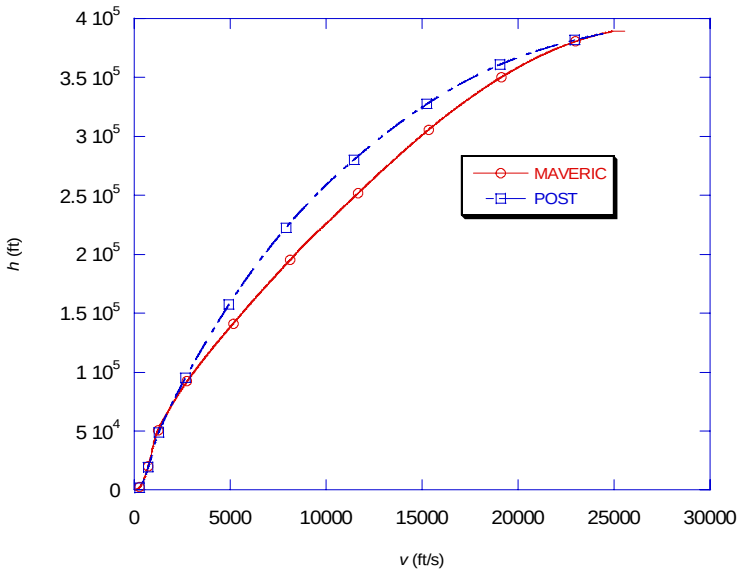

Figure 4-16: Altitude vs Speed, POST/MAVERIC Comparison, 51.6 deg mission

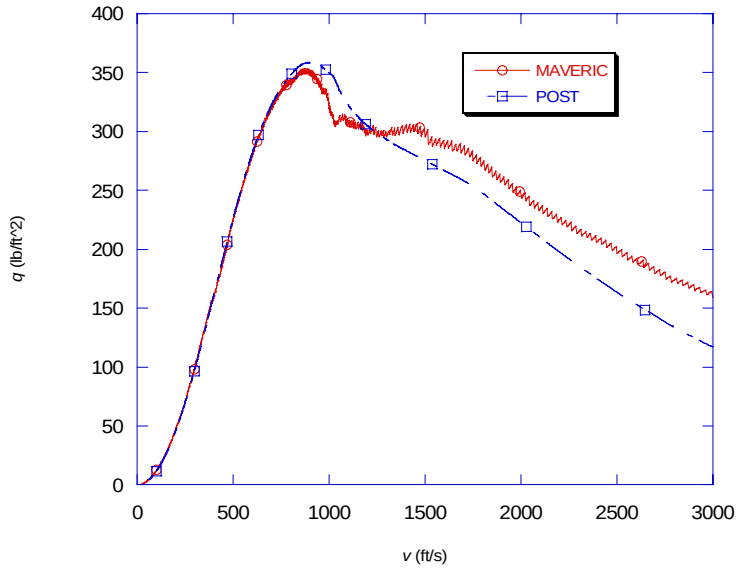

Figure 4-17: Dynamic Pressure vs Speed, POST/MAVERIC Comparison, 51.6 deg mission

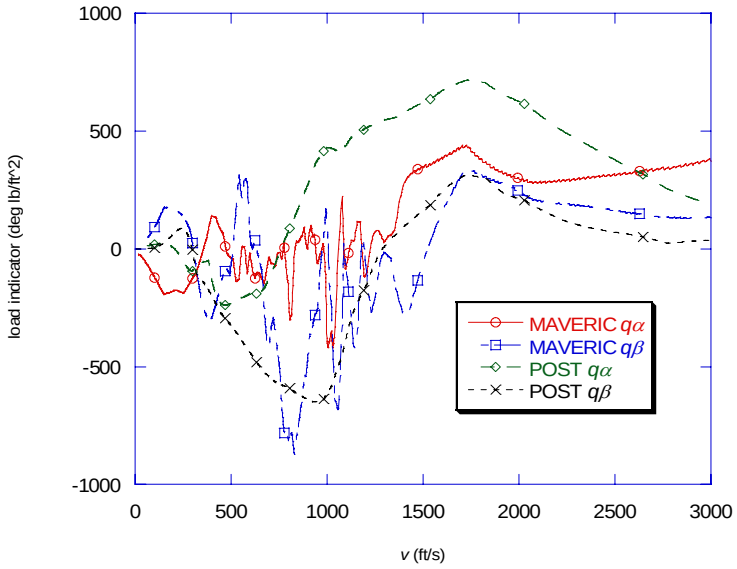

Figure 4-18: $q\alpha$, $q\beta$ vs Speed, POST/MAVERIC Comparison, 51.6 deg Mission

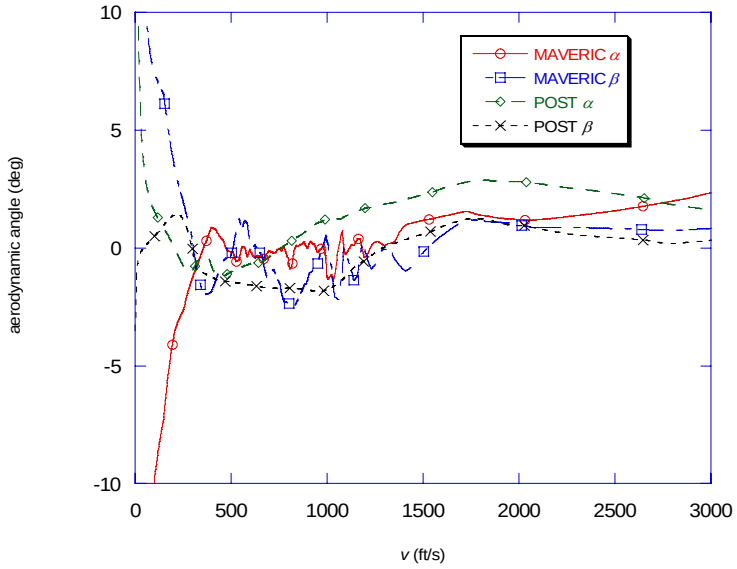

Figure 4-19: α, β vs Speed, POST/MAVERIC Comparison, 51.6 deg Mission

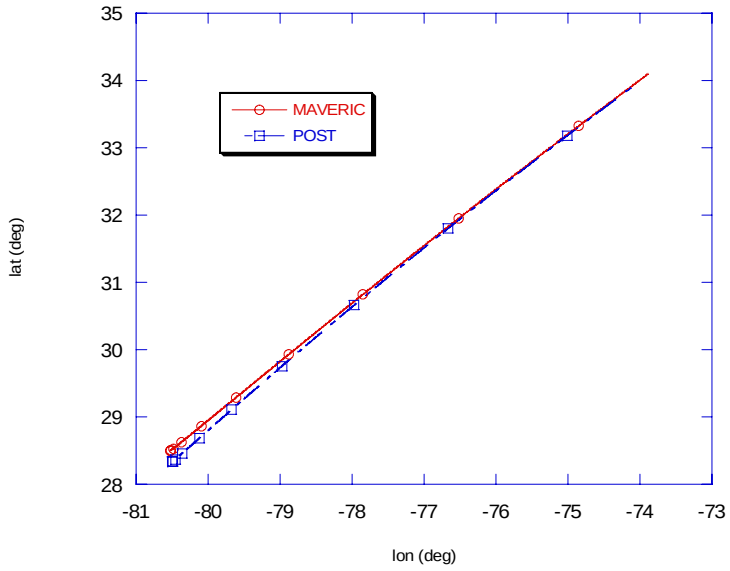

Figure 4-20: Ground Track, POST/MAVERIC Comparison, 51.6 deg Mission

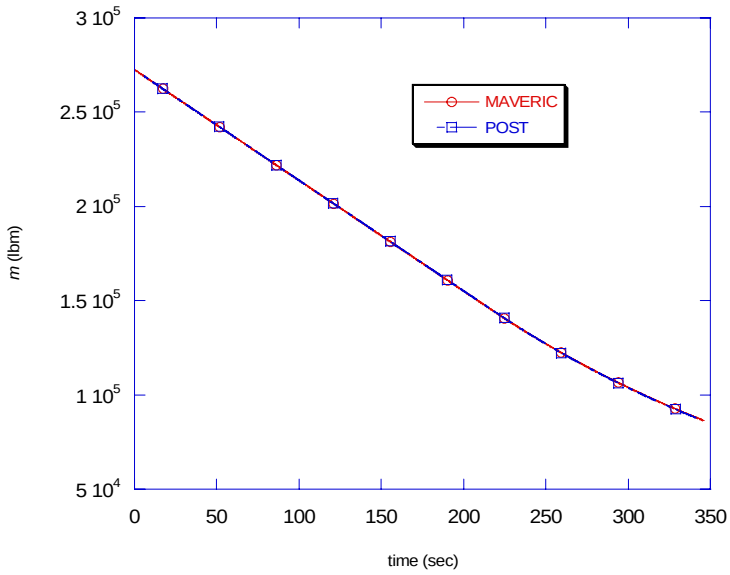

Figure 4-21: Mass vs Time, POST/MAVERIC Comparison, 51.6 deg Mission

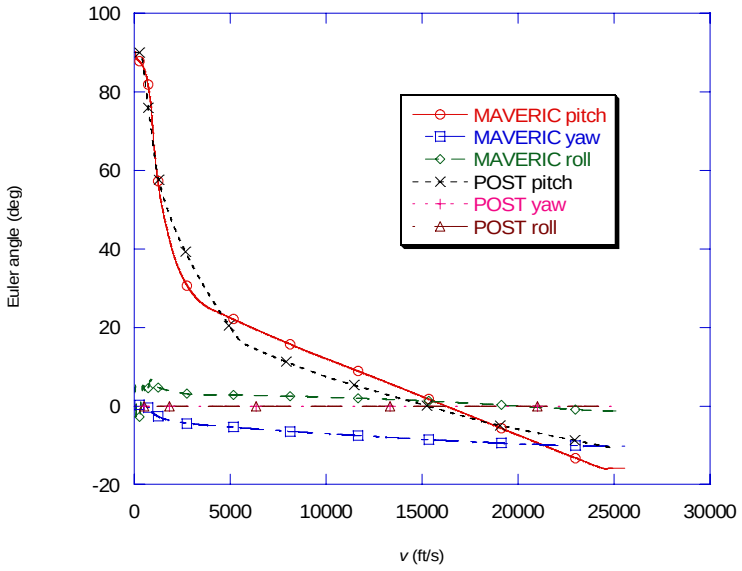

Figure 4-22: Euler Angles vs Speed, POST/MAVERIC Comparison, 51.6 deg Mission

CHAPTER 5

TRANSVERSALITY CONDITIONS

The transversality conditions of optimality serve to help determine the optimal values of the initial costates. In works by Calise et al. [25,28], Dukeman [46], and Lu et al. [41], analytical expressions for the vector of constant parameters, ν, are determined a priori and applied to obtain a reduced set of transversality conditions. This procedure is performed for each set of terminal state constraints. If a trajectory with a new set of constraints is to be optimized, then the analytical process of determining the reduced set of transversality conditions must be repeated. This is the case even when a new set is similar to an existing set. For example, the reduced transversality conditions for the case of constrained apogee and perigee is significantly different from the case of constrained apogee, perigee, and true anomaly, and so on. The consequence is a lack of flexibility in being able to compute trajectories with different mission objectives. What is desired is a numerical method of treating transversality conditions so that the guidance practitioner can specify the particular terminal state constraints that apply for a given mission and let the algorithm handle the problem of determining the reduced transversality conditions numerically. This chapter presents a method for doing this which is found to work well in practice. The results presented in Chapters 4, 6 and 7 use this method. A menu of over 20 constraints was developed allowing the user to combine terminal constraints as necessary to define mission objectives. As usual, it is required that the specified constraints be linearly independent at the terminus of the powered maneuver. This in turn requires sufficient knowledge

of flight/orbital mechanics on the part of the user. As a prelude to the numerical treatment, analytical methods for reducing the transversality conditions are discussed.

5.1 Analytical Treatment

Consider the performance index

$$J = \phi\left(x\left(t_f\right)\right) + \int_{t_0}^{t_f} L\left(x, u, t\right) dt \tag{5-1}$$

subject to the terminal state constraints (2-4). Adjoining the terminal state constraints with constant multipliers $\boldsymbol{\nu}$ to the performance index results in

$$J = \left[\phi + \nu^T \boldsymbol{\psi}\right]_{t=t_f} + \ldots \tag{5-2}$$

Taking the variation of the augmented performance index and integrating by parts results in (see [7])

$$dJ = \left[\left(\phi_x + \boldsymbol{\nu}^T \boldsymbol{\psi}_{\mathbf{x}} - \boldsymbol{\lambda}^T\right)\mathbf{dx}\right]_{t=t_f} + \ldots \tag{5-3}$$

Setting the variation to zero and treating the variation of final state, **dx**, as an arbitrary variation, the necessary conditions known as transversality conditions result [7,44]:

$$\left[\boldsymbol{\lambda}^{\mathbf{T}} = \varphi_x + \boldsymbol{\nu}^T \boldsymbol{\psi}_{\mathbf{x}}\right]_{t=t_f} \tag{5-4}$$

For an alternative interpretation of the transversality conditions, note that there exist $6 - k$ linearly independent 6-vectors, $\mathbf{a}_i$, that satisfy

$$\boldsymbol{\psi}_{\mathbf{x}} \mathbf{a}_i = \mathbf{0}, \quad i = 1,2,\ldots,6-k \tag{5-5}$$

where $\boldsymbol{\psi}_{\mathbf{x}}$ is the matrix whose m^{th} row is the gradient of the m^{th} terminal state constraint with respect to state. Taking the inner product of each side of (5-4) with the $\mathbf{a}_i$ results in the alternate form of the transversality conditions:

$$\left(\boldsymbol{\lambda}^T - \phi_x\right)\mathbf{a}_i = 0, \quad i = 1,2,\ldots,6-k \tag{5-6}$$

The $\mathbf{a}_i$ form a set of basis vectors for the null space of $\boldsymbol{\psi}_{\mathrm{x}}$. An alternative statement of the transversality conditions, then, is that the final costate vector minus the gradient of the terminal cost must be orthogonal to the null space of $\boldsymbol{\psi}_{\mathrm{x}}$. Note that this eliminates the constant multipliers $\boldsymbol{\nu}$.

Now, starting again from the performance index (5-1), this time electing not to adjoin the terminal state constraints to the performance index, take the first variation, integrate by parts and set to zero to obtain the transversality conditions of the form

$$\left[\left(\boldsymbol{\lambda}^{\mathbf{T}} - \varphi_{\mathbf{x}}\right)\mathbf{dx}\right]_{t=t_f} = 0 \tag{5-7}$$

Here, the only state variations, **dx**, that need to be considered are admissible state variations, that is, the state variations that satisfy, to first order, the terminal state constraints

$$\begin{aligned} \mathbf{0} &= \boldsymbol{\psi}(\mathbf{x}+\mathbf{dx}) = \boldsymbol{\psi}(\mathbf{x}) + \boldsymbol{\psi}_x \mathbf{dx} \\ &= \mathbf{0} + \boldsymbol{\psi}_x \mathbf{dx} \\ &\Rightarrow \boldsymbol{\psi}_x \mathbf{dx} = \mathbf{0} \end{aligned} \tag{5-8}$$

where **x** is any state vector that satisfies the terminal state constraints. Apparently the admissible state variation vectors are also in the null space of $\boldsymbol{\psi}_\mathbf{x}$. The admissible state variation interpretation of the transversality conditions is more meaningful than the null space basis vectors interpretation and it's often easier to analytically derive expressions for admissible state variations than it is to analytically obtain null space basis vectors. Imposition of k terminal state constraints ψ_i, i=1,2,…,k, for a particular mission results in a dimension of 6 - k for the admissible state variation subspace. Thus, there are 6 - k linearly independent variational vectors **dx** and they span the null space of $\boldsymbol{\psi}_\mathbf{x}$ as do the $\mathbf{a}_i$ vectors of (5-6). The transversality conditions then are expressed as

$$\left(\boldsymbol{\lambda}^T - \boldsymbol{\varphi}_\mathbf{x}\right)\mathbf{dx}_i = 0, \quad i = 1,2,\ldots,6-k \tag{5-9}$$

Note in (5-9) the direction of the **dx** vectors is what is important, i.e., any vector **dx** that satisfies (5-9) can be scaled arbitrarily in magnitude and still satisfy (5-9). Derivations of several admissible state variation vectors corresponding to particular quantities of interest are given below.

Free true anomaly:

An unconstrained true anomaly corresponds to a coast in a Keplerian orbit; hence, the admissible position and velocity increments are

$$\begin{aligned} \mathbf{dr} &= \varepsilon \mathbf{v} \\ \mathbf{dv} &= -\varepsilon\mu \frac{\mathbf{r}}{\|\mathbf{r}\|^3} \end{aligned} \tag{5-10}$$

where $\varepsilon > 0$ is used to indicate that the perturbations are small. Thus, the admissible state variation direction corresponding to a free true anomaly is

$$\mathbf{dx}_\nu^T = \left(\mathbf{v}^T, -\mu \frac{\mathbf{r}^T}{\|\mathbf{r}\|^3} \right) \tag{5-11}$$

Another way to see this is to consider a mission where all orbital elements are fixed except for the true anomaly. Thus, all terminal state constraints are constants of the (coasting) motion so that, for each terminal state constraint

$$0 = \dot{\psi} = \boldsymbol{\psi}_\mathbf{x} \dot{\mathbf{x}} = \boldsymbol{\psi}_\mathbf{x} \left(\mathbf{v}^T, -\mu \frac{\mathbf{r}^T}{\|\mathbf{r}\|^3} \right)^T \Rightarrow \boldsymbol{\psi}_\mathbf{x} \left(\mathbf{v}^T, -\mu \frac{\mathbf{r}^T}{\|\mathbf{r}\|^3} \right)^T = 0 \tag{5-12}$$

Recalling that any non-zero vector in the null space of ψ_x is also an admissible state variation, (5-11) is verified. Note that this vector is an admissible state variation for any mission whose terminal state constraints consist of orbital constants only (see also Appendix C.)

Free argument of perigee:

Consider small position and velocity increments, in the orbital plane, and normal to the position and velocity vectors, respectively. When these increments are added to the position and velocity, respectively, the perturbed position and velocity vectors are the same as those vectors that would result from simply rotating the position and velocity vectors about the vehicle's angular momemtum vector. Among the classical orbital elements, this rotation perturbs only the argument of perigee. Mathematically, the position and velocity increments are expressed as [44]:

$$\begin{aligned} \mathbf{dr} &= \mathbf{r} \times \varepsilon \mathbf{h} \\ \mathbf{dv} &= \mathbf{v} \times \varepsilon \mathbf{h} \end{aligned} \tag{5-13}$$

Note that the position and velocity are rotated through the same angle. Thus, the admissible state variation direction corresponding to free argument of perigee is:

$$\mathbf{dx}_{\omega}^{T} = \left[(\mathbf{r} \times \mathbf{h})^{T}, (\mathbf{v} \times \mathbf{h})^{T} \right] \tag{5-14}$$

Free longitude of the ascending node:

Free longitude of the ascending node corresponds to a free rotation of the position and velocity vectors about the polar axis, $\mathbf{U_p}$, hence:

$$\begin{aligned} \mathbf{dr} &= \mathbf{r} \times \varepsilon \mathbf{U_p} \\ \mathbf{dv} &= \mathbf{v} \times \varepsilon \mathbf{U_p} \end{aligned} \tag{5-15}$$

Thus, the admissible state variation direction corresponding to free node is:

$$\mathbf{dx}_\Omega^T = \left[\left(\mathbf{r} \times \mathbf{U_p} \right)^T, \left(\mathbf{v} \times \mathbf{U_p} \right)^T \right] \tag{5-16}$$

Note that the unit Polar axis, $\mathbf{U_p}$, must of course be expressed in the same frame as **r** and **v**.

Free inclination:

Free inclination corresponds to a free rotation about the node vector, hence:

$$\begin{aligned} \mathbf{dr} &= \mathbf{r} \times \varepsilon \left(\mathbf{U_p} \times \mathbf{h} \right) = \varepsilon \left[\left(\mathbf{r} \cdot \mathbf{h} \right) \mathbf{U_p} - \left(\mathbf{r} \cdot \mathbf{U_p} \right) \mathbf{h} \right] = -\varepsilon r_z \mathbf{h} \\ \mathbf{dv} &= \mathbf{v} \times \varepsilon \left(\mathbf{U_p} \times \mathbf{h} \right) = \varepsilon \left[\left(\mathbf{v} \cdot \mathbf{h} \right) \mathbf{U_p} - \left(\mathbf{v} \cdot \mathbf{U_p} \right) \mathbf{h} \right] = -\varepsilon v_z \mathbf{h} \end{aligned} \tag{5-17}$$

where $\mathbf{U_p}$ is a unit vector parallel to the North Pole. Thus, the admissible state variation direction corresponding to free inclination is:

$$\mathbf{dx}_i^T = \left[-r_z \mathbf{h}^T, -v_z \mathbf{h}^T \right] \tag{5-18}$$

Note that the previous equation is valid for components expressed in any given inertial frame because the dot products in (5-17) evaluate to the same value regardless of coordinate frame.

Free Position on a Circle Centered at a Heading Alignment Cone:

For the case of abort to a landing site, the (great circle) range to the landing site is constrained through altitude, speed, and flight path angle versus range relationships. The heading error is constrained to be zero. This constraint set allows a small rotation about the landing site position vector $\mathbf{r}_{\mathbf{HAC}}$.

$$\begin{aligned} \mathbf{dr} &= \mathbf{r} \times \varepsilon \mathbf{r}_{\mathbf{HAC}} \\ \mathbf{dv} &= \mathbf{v} \times \varepsilon \mathbf{r}_{\mathbf{HAC}} \end{aligned} \tag{5-19}$$

Thus, the admissible state variation corresponding to free position on a circle centered at a HAC is:

$$\mathbf{dx}_{HAC}^{T} = \left[\left(\mathbf{r} \times \mathbf{r}_{\mathbf{HAC}} \right)^{T}, \left(\mathbf{v} \times \mathbf{r}_{\mathbf{HAC}} \right)^{T} \right] \tag{5-20}$$

Free Down Range, Altitude, Speed Functions of Range

For the case of abort to a landing site, the final range is free but the altitude and speed are constrained functions of range, *R,* through the given entry profile. The corresponding free state variations are:

$$d\mathbf{x} = \frac{\partial \mathbf{x}}{\partial R} dR = \begin{bmatrix} \dfrac{\partial (r\hat{\mathbf{r}})}{\partial R} \\ \dfrac{\partial (v\hat{\mathbf{v}})}{\partial R} \end{bmatrix} dR = \begin{bmatrix} \dfrac{\partial r}{\partial R}\dfrac{\mathbf{r}}{r} + \dfrac{r}{R_e}\dfrac{(\mathbf{r}\times\mathbf{r}_{HAC})\times\mathbf{r}}{|(\mathbf{r}\times\mathbf{r}_{HAC})\times\mathbf{r}|} \\ \dfrac{\partial v}{\partial R}\dfrac{\mathbf{v}}{v} - v\dfrac{(\mathbf{r}\times\mathbf{r}_{HAC})\times\mathbf{v}}{|(\mathbf{r}\times\mathbf{r}_{HAC})\times\mathbf{v}|}\left(-\dfrac{1}{R_e}\right) \end{bmatrix} dR \tag{5-21}$$

Many useful sets of terminal state constraints and their associated transversality vectors have been derived in Refs. [8,45].

5.2 Numerical Treatment

At the final time, t_f, k terminal state constraints are prescribed:

$$\psi_i = \psi_i(\mathbf{r}(t_f), \mathbf{v}(t_f)) = 0,\ i = 1,\ldots,k,\ \ k \le 6 \tag{5-22}$$

The transversality conditions are given by [7]

$$\boldsymbol{\lambda}(t_f) - \varphi_{\mathbf{x}}^{\mathrm{T}}(\mathbf{x}(t_f), t_f) = \boldsymbol{\psi}_{\mathbf{x}}^{T}(\mathbf{x}(t_f), t_f)\boldsymbol{\nu} \tag{5-23}$$

where φ is the 'terminal cost' (which may or may not be present for a given problem), $\boldsymbol{\psi}$ is a column vector composed of the k terminal state constraints, $\boldsymbol{\nu}$ is a vector composed of k arbitrary constants used to adjoin the terminal state constraints to the cost, and the subscript $\mathbf{x}$ denotes the gradient of the designated quantity with respect to the state vector. In much of the past guidance research, the transversality conditions (5-23) are treated analytically in the sense that $\boldsymbol{\nu}$ is eliminated a priori by solving in terms of final state/costate, leaving a reduced set of $6 - k$

transversality conditions. There are several different but equivalent approaches ([8], [25], [41], and [46], among others) for solving for $\boldsymbol{\nu}$.

The original motivation for this modus operandi was computation speed [8] which was in short supply when some of the earliest guidance research using optimal control theory was performed. Note that if any one of the k constraints is removed or if a different constraint is added, then each component of $\boldsymbol{\nu}$ changes thus requiring a repeat of the analyses to solve for $\boldsymbol{\nu}$. In other words, each unique set of terminal state constraints has a unique solution for $\boldsymbol{\nu}$, e.g., the constraint set wherein node and inclination are free results in a different solution for $\boldsymbol{\nu}$ than the constraint set wherein inclination is prescribed but node is free, etc. If it is desired to experiment with several different constraint sets, the requirement that $\boldsymbol{\nu}$ be solved a priori quickly becomes a burden.

What is needed is a numerical method. The simplest approach would be to lump the transversality conditions (5-23) with all the other boundary conditions (i.e., terminal state constraints, switching conditions, continuity conditions, etc.) so that the solution for $\boldsymbol{\nu}$, along with all the other independent problem variables (initial costates, switching times, etc.) would be accomplished via Newton's method. However, this adds to the size of the associated Jacobian matrix and the cost of solving the linear system is proportional to n^3 where n is the number of equations (parameters), so it is preferable to avoid lumping more equations in with our boundary conditions if possible. Instead, k of the six equations of (5-23) are chosen to solve for $\boldsymbol{\nu}$. The question that immediately arises is which k equations to use? As long as the gradients matrix, $\boldsymbol{\psi}_{\mathbf{x}}$, is full rank, then there exists at least one non-singular $k \times k$ sub-matrix of the gradients matrix that can be used to solve for $\boldsymbol{\nu}$. A few very simple modifications to, say, a conventional Gauss or LU-decomposition linear solver code, can be made so that a system of m equations in n unknowns ($m \geq n$) like that of (5-23) (where $m = 6$, $n = k \leq 6$) can be solved for the n unknowns using any n of the equations. Conventional pivoting logic determines a non-singular k by k sub-matrix within $\boldsymbol{\psi}_{\mathbf{x}}$

from which ν is solved. The code is designed to report which k of the 6 equations are used so that the $6 - k$ equations that are not used can be lumped in with all the boundary conditions to do their part in determining the initial costate values.

Numerical treatment of the transversality conditions allows us to develop into the guidance algorithm a menu of functions that can be used interchangeably as terminal cost functions or as terminal state constraints. Users of the algorithm can then mix and match terminal state constraints and cost functions at will without re-analyzing transversality conditions for the particular combination specified. Examples of useful functions include apogee, inclination, radius magnitude, speed, orbital energy, heading error to a landing site, etc.

The numerical results presented in Chapters 4, 6 and 7 use the numerical approach outlined above.

CHAPTER 6

FREE FINAL TIME MULTI-ARC OPTIMIZATION FORMULATION

The guided trajectory results presented in Chapter 4 were obtained using a single-burn optimization formulation. Consider the possibility of allowing for a coast arc inserted in the middle of a trajectory so that the optimized trajectory consists of a burn-coast-burn, also "multi-arc," transfer. In principle, we've added two new parameters to the problem, the beginning and end times of the coast. Therefore, one would expect the fuel expenditure in the burn-coast-burn case to be less than or equal to that of the single burn case. Of course, the single burn transfer is just a special case of the burn-coast-burn transfer with coast duration of zero.

It is useful to anticipate mission scenarios in which there might be observed significant benefits in allowing multi-arc trajectories. Because velocity changes can be made more directly, via thrust, than can position changes, it is reasonable to expect transfers requiring a change in position to benefit from a coast, with the amount of benefit proportional to the amount of required position change. During the coast, the position evolves under the influence of gravity and inertial effects, while incurring no additional cost. In the case of large acceleration due to thrust, the burns look like impulsive burns, with the first burn setting up the coast initial conditions so that the vehicle coasts to where it needs to be, and then the second burn provides what looks like an impulsive ΔV to effect the final needed velocity change. An example is the burn-coast-burn Hohmann transfer between two circular orbits. The circle-to-circle transfer's potential energy change is twice that of the kinetic energy change meaning more energy is needed to change the position than the velocity. The larger the difference between the initial and final circular radii,

the more dramatic is the benefit of performing a burn-coast-burn transfer instead of a single burn transfer. Likewise, higher acceleration vehicles benefit more from allowing a coast than do low acceleration vehicles. For the ascent problem, one would expect, e.g., launches into high circular orbits to benefit more from burn-coast-burn protocol than would launches into lower circular orbits. Because the cost rate is so high during a burn for high acceleration vehicles, it makes sense to allow a coast so that gravity and inertial effects can effect a portion of the position change. Conversely, in case of very low thrust, the circle-to-circle transfer is accomplished adequately via one long continuous burn which maintains the vehicle in a near-circular orbit as the radius gradually changes to the target value.

The energy change in the Earth-to-orbit problem is dominated by the kinetic energy change. An approximate linear analysis reveals that for a 150 n.m. destination orbit, the kinetic energy change is approximately 10 times that of the potential energy change. This fact, combined with the relatively long burn time required to impart the kinetic energy change, i.e., the burn doesn't look like an impulsive burn, leads us to predict that a dramatic ΔV reduction will not be realized by flying a burn-coast-burn trajectory instead of a single burn. However, because the sensitivity of injected mass to required ΔV is so high for the Earth-to-orbit problem, a noticeable and welcome improvement in payload capability would nevertheless be expected.

This line of reasoning suggests that, for the case of downrange aborts, discussed in Chapters 8 and 9, the abort trajectory can be stretched by allowing a burn-coast-burn sequence. To avoid the undesirable act of turning off and restarting engines, the burn-coast-burn trajectory can perhaps be approximated by a max thrust-min thrust-max thrust sequence.

This chapter derives strengthened necessary conditions for the multi-arc problem. The necessary conditions of [8,25,42] are valid for certain sets of terminal state constraints, but not others, as described in the sequel. A modified form of multiple shooting is described as a means of solving for the optimized burn-coast-burn trajectory. Open-loop optimization results for a

two-stage launch vehicle are presented using the described formulation and numerical methods. Chapter 7 expands the present discussion to a closed-loop guidance implementation and shows guided trajectory results.

6.1 General Development

The objective of the problem is to minimize the fuel usage or, equivalently, maximize the vehicle mass. The performance index to be maximized is:

$$J = \int_{t_0}^{t_f} \dot{m} dt = \int_{t_0}^{t_f} -\frac{T}{c} dt, \qquad 0 \le T \le T_{\max} \tag{6-1}$$

The Hamiltonian is given by:

$$H = \boldsymbol{\lambda}_{\mathbf{r}}^{\mathrm{T}} \mathbf{v} + \boldsymbol{\lambda}_{\mathbf{v}}^{\mathrm{T}} \left[\mathbf{g}(\mathbf{r}) + \frac{T}{m} \frac{\boldsymbol{\lambda}_{\mathbf{v}}}{|\boldsymbol{\lambda}_{\mathbf{v}}|} \right] - \lambda_m \frac{T}{c} - \frac{T}{c} \tag{6-2}$$

Let H_{NT} denote that part of the Hamiltonian that is independent of thrust and define the switching function, S, as the partial derivative of the Hamiltonian with respect to thrust, so that the Hamiltonian is re-written as:

$$H = \boldsymbol{\lambda}_{\mathbf{r}}^{\mathrm{T}} \mathbf{v} + \boldsymbol{\lambda}_{\mathbf{v}}^{\mathrm{T}} \mathbf{g}(\mathbf{r}) + T \left(\frac{|\boldsymbol{\lambda}_{\mathbf{v}}|}{m} - \frac{\lambda_m}{c} - \frac{1}{c} \right) \equiv H_{NT} + TS \tag{6-3}$$

The Hamiltonian is apparently maximized according to the following rule:

$$\begin{aligned} T &= T_{\max,} \qquad S > 0 \\ T &= 0, \qquad\quad S < 0 \end{aligned} \tag{6-4}$$

For the case of free final time, t_f, it is well known that on an optimal trajectory, the Hamiltonian is identically zero:

$$H(t) = 0, \qquad t_0 \le t \le t_f \tag{6-5}$$

At each (thrust on/off) switching point, t_i, $S = 0$, hence $H_{NT} = 0$. However, this only gives us about half the necessary conditions needed to determine the switching times because H_{NT} is identically constant on a coast arc. It is required that S be zero at the beginning and end of each coast arc, thus $S(t_i) - S(t_{i-1}) = 0$. However, the mass and the mass costate are constant on a coast arc. Thus, the condition reduces to a requirement that the magnitude of the primer vector at the beginning of the coast equal that at the end of the coast. The interior point necessary conditions then are [8,42]:

$$\begin{aligned} &\left|\boldsymbol{\lambda}_{\mathbf{v}}(t_i)\right| - \left|\boldsymbol{\lambda}_{\mathbf{v}}(t_{i-1})\right| = 0, \quad \text{across a coast arc} \\ &H_{NT}(t_{i+1}) - H_{NT}(t_i) = 0, \quad \text{across a burn arc} \end{aligned} \tag{6-6}$$

The main reason for using (6-6) instead of the conditions $S(t_i) = 0$, $i = 1,2,\ldots,n$, is that use of (6-6) eliminates the need to numerically integrate the mass costate. Despite not having mass costate available from the optimization process, it can still be determined whether the addition of a coast arc will improve the performance. Using the condition $H(t) = 0$ results in $S(t) = -H_{NT} / T$ which

means that S and H_{NT} are always of opposite sign on an optimal trajectory. If H_{NT} is positive anywhere on a burn arc, then S is negative in violation of (6-4) so a coast arc insertion would improve performance.

One limitation of the previous set of necessary conditions is that there is an inherent assumption made by the authors of [8] and [42] that $H_{NT}(t_f) = 0$. The latter condition is true as long as the prescribed terminal constraints are constants in the absence of thrust, e.g., semi-major axis, perigee, etc., and this is commonly the case, but in general $H_{NT}(t_f)$ is non-zero. An alternative approach is to initialize the switching function at the first (free) switching point, say, t_1, in the trajectory using the fact that the Hamiltonian is zero along the optimal trajectory:

$$S(t_1) = -\frac{H_{NT}(t_1)}{T} \tag{6-7}$$

where T is the non-zero value of thrust just before, or just after the first switching time as appropriate. If the next switching point, t_2, corresponds to the end of a coast, then, using the fact that the mass costate and vehicle mass are constants along a coast arc, the value of the switching function at the end of the coast arc, t_2, is given by:

$$S(t_2) = S(t_1) + (S(t_2) - S(t_1)) = S(t_1) + \frac{|\lambda_v(t_2)| - |\lambda_v(t_1)|}{m} \tag{6-8}$$

whereas if t_2 corresponds to the end of a burn arc, using the fact that the Hamiltonian is identically constant across a burn arc, the switching function at t_2 is given by:

$$S(t_2) = S(t_1) + \frac{H_{NT}(t_1) - H_{NT}(t_2)}{T} \tag{6-9}$$

The switching function can be computed at each switching point in this manner enabling application of the interior point necessary conditions:

$$S(t_i) = 0, \qquad i = 1,2,\ldots,n \tag{6-10}$$

where n is the number of free switching points. These conditions are equivalent to those of [8,42] except that this set ensures that the value of the switching functions at each switching point is zero as compared with merely requiring that

$$S(t_i) = -H_{NT}(t_f), \qquad i = 1,2,\ldots,n \tag{6-11}$$

where $H_{NT}(t_f)$ is not necessarily equal to zero. Note that the necessary conditions given here differ from those of [25] where the final time was fixed and application of the interior point necessary conditions required backwards calculation of the switching function starting at t_f.

As an application, consider a launch into orbit with a fixed amount of propellant available for the main engines and a separate tank of fuel available for the orbital maneuvering system (OMS) engines. Once the main engines are turned off, it is undesirable to restart them. Thus, it is optimal to burn the main engine propellant to depletion during the first burn, followed by a coast arc, the duration of which is to be optimized. At the conclusion of the coast arc, the OMS engines are used to circularize the orbit at some target altitude. One (interior) switching time is free, that is, the start of the OMS burn. In this case, the necessary condition for determining the OMS burn start time, t_1, reduces simply to:

$$H_{NT}(t_1) = 0 \tag{6-12}$$

6.2 Multiple Shooting Method Applied to Multi-Arc Optimization

The burn-coast-burn (indirect) optimization problem is very sensitive in that very small perturbations in the initial costates can cause large changes in the trajectory. The multiple shooting method [40] is a proven method for mitigating initial guess sensitivity. The implementation of multiple shooting requires a strategy for placement of the nodes. It turns out that a reasonable and convenient node placement strategy for the multi-burn optimization problem is to introduce a node at each thrust switching time, i.e., at the boundaries that separate a burn from a coast arc. The following describes how guesses for the state/costate vector and switching times are constructed at each shooting point.

The first shooting point corresponds to main engine cutoff and the time at which this occurs is not a free parameter to be optimized because it is a function of main propulsion propellant loading (it is assumed that this propellant is burned to depletion.) A reasonable guess for the state at MECO is provided by assuming that the vehicle burns out at a perigee several tens of kilometers below the target radius and with an apogee equal to the target radius. Thus, the guessed speed and radius at MECO are:

$$r_{MECO} = r_p$$

$$v_{MECO} = \sqrt{2\mu\left(\frac{1}{r_{tgt}} - \frac{1}{r_{tgt} + r_p}\right)} \tag{6-13}$$

where r_p is the guess for MECO radius/perigee. In the guidance frame (x-axis up, z-axis downrange) then, the position and velocity guesses at MECO are:

$$\mathbf{r}_{MECO} = \begin{bmatrix} r_p \\ 0 \\ 0 \end{bmatrix} \quad \mathbf{v}_{MECO} = \begin{bmatrix} 0 \\ 0 \\ v_{MECO} \end{bmatrix} \tag{6-14}$$

The guess for the start time of the second burn is obtained by assuming that the second burn (i.e., OMS burn) will start at the apogee of the transfer orbit, hence, half the period of the transfer orbit is added to the calculated MECO time. The position and velocity guesses at the start of the OMS burn are opposite in direction to the position and velocity vectors at MECO:

$$\begin{aligned} \mathbf{r}_{OMS} &= -\frac{r_{tgt}}{r_p}\mathbf{r}_{MECO} \\ \mathbf{v}_{OMS} &= -\frac{v_{OMS}}{v_{MECO}}\mathbf{v}_{MECO} \end{aligned} \tag{6-15}$$

where the speed at OMS ignition, $v_{OMS,}$ is given by:

$$v_{OMS} = \sqrt{2\mu\left(\frac{1}{r_{tgt}} - \frac{1}{r_{tgt}+r_p}\right)} \tag{6-16}$$

Once the velocity guesses at the nodes are constructed, the primer vector guesses are set equal to the respective unitized velocity vectors and the remaining components of the costate vector are set to small non-zero values.

The burn coast burn problem solved via multiple shooting has associated with it: 1) 12 continuity conditions at each of two nodes to determine the 12 components of the state/costate at each of those nodes, 2) one switching condition (from optimality) at the second node to determine the start of the OMS burn, 3) k prescribed terminal state constraints plus 6 - k transversality conditions (from optimality) to determine the 6 components of the initial costate, and 4) one 'non-constraining' constraint on the magnitude of the final primer vector to determine the final time. In total then, there are 32 constraints to determine 32 independent variables. Applying

Newton's method to solve for the independent variables thus results in a linear system with a 32 by 32 matrix. The associated Jacobian matrix is sparse (i.e., most of the entries are zero) because any given independent variable may affect constraints at two nodes at most. It is computationally wasteful to solve sparse linear systems using, say, conventional Gaussian elimination or LU-decomposition because most of the computational effort would involve manipulating and doing arithmetic on zero entries. Iterative methods such as the conjugate-gradient method are well suited to solving sparse systems but they are complex methods and are better suited for large linear sparse systems where the matrices have dimension of, say, 100 or more.

It turns out that the Jacobian matrix, if the constraints and independent variables are grouped by node, is of a special form referred to as band diagonal. In particular, there exists integers m_1, $m_2 < n$, the number of equations, such that any matrix entry m_1+ 1 rows/columns below/left of the main diagonal is zero and, likewise, any matrix entry m_2 + 1 rows/columns above/right of the main diagonal is zero. There are simple and efficient linear system solver algorithms that avoid manipulating and performing arithmetic on the zero entries of the matrix. The band diagonal version of the LU-decomposition algorithm [38] is algorithmically simple and has been coded (via [38]) for use in the present problem. In order to use this algorithm, it must be supplied with the values of m_1, m_2. The particular values of m_1 and m_2 depend on the problem being solved, i.e., how many nodes there are, whether or not there exist free interior switching times, etc. However, the code that determines m_1 and m_2 does not need to 'know' anything specific about the problem being solved - m_1 and m_2 are determined in the code simply by comparing row and column indices during the construction of the Jacobian matrix.

6.3 Numerical Results

A generic 2-stage RLV (see vehicle data in Appendix E) ascent to 28.5 deg inclination, 150 n.m. circular orbit, starting at 1st stage burn out, was optimized as a burn-coast-burn mission sequence, using the optimization theory described in the prequel and solved using multiple shooting. Figure 6-1 shows the base 10 logarithm of the residuals associated with the necessary conditions and the continuity conditions. The residuals, in order from one to 32, correspond to position component continuity (km units), velocity component continuity (km/s units), primer vector continuity, control rate continuity, all at the first node (i.e., at end of the first burn), followed by 12 analogous quantities at the second node (i.e., at the start of the OMS burn), the switching condition at the start of the OMS burn, 4 terminal state constraints, 2 transversality constraints, and finally, the primer magnitude constraint. Note, for example, that the position component continuity residuals (numbers 1,2,3,13,14, and 15) are all less than a thousandth of a meter and, likewise, all other continuity residuals are very small. Being able to drive the continuity residuals to very small values is desirable in that it improves our confidence that the obtained solution is valid (i.e., satisfies all the governing differential equations over the entire mission time interval). The switching condition, boundary condition number 25, is on the order of ten to the minus seventeen (on the initial guess trajectory, it was about 13 orders of magnitude larger than the converged value, so this gives us confidence that the algorithm has indeed driven the switching condition to a value that is sufficiently close to zero. The terminal state constraints and the transversality conditions are likewise small enough to give us confidence that the desired optimal trajectory has been found.

Figure 6-2 shows the speed vs. time on the optimized trajectory. The end of the first burn occurs at 436.7 s, the start of the second burn, at 1209.1 s, the end of the second burn, at 1403.4 s. Most of the velocity addition is provided by the main propulsion system, as expected. Figure 6-3

shows altitude versus time with altitude monotonically increasing throughout the trajectory. The vehicle depletes the main propulsion system propellant (at about 350,000 feet altitude), coasts to apogee, and performs a short burn to circularize the orbit. Figure 6-4 shows apogee and perigee vs. time. The second burn is almost entirely a perigee-raise burn with apogee unaffected. Figure 6-5 shows flight path angle vs. time with MECO occurring at about 2.5 degrees. Figure 6-6 shows total vehicle mass vs. time and illustrates that the second burn is a low-thrust burn (with acceleration of approximately 0.1 g's). Figure 6-7 shows the switching function, S, vs. time. Figure 6-8 shows the switching function during the second burn. Recall that during burns, S should be > 0, else insertion of a coast arc would improve the solution. Figures 6-7 and 6-8 show that S is > 0 during both burns and, hence, this particular two-burn solution cannot be improved upon with insertion of a coast arc.

Note from Figure 6-8 that H_{NT} is (numerically) equal to zero at the final time even though H_{NT} = 0 is not an explicitly prescribed constraint. The explanation for this is that, for an optimal trajectory, the final costate vector is orthogonal to the vector subspace constructed from the set F of all state perturbations $\delta\mathbf{x}^{\mathrm{T}} = [\,\delta\mathbf{r}^{\mathrm{T}}, \delta\mathbf{v}^{\mathrm{T}}]$ that satisfy the k terminal state constraints ψ_i, that is:

$$\boldsymbol{\lambda}(t_f) \perp F, \quad F = \left\{\boldsymbol{\delta}\mathbf{x} : \psi_i\left(\mathbf{x}^{opt}(t_f) + \boldsymbol{\delta}\mathbf{x}\right) = 0, i = 1,2,\ldots,k\right\} \tag{6-17}$$

Equation (6-17) is just a restatement of the transversality conditions (2-4). For the circular orbit mission, position and velocity perturbations in the velocity and gravity directions, respectively, are admissible state perturbations. In other words, the final position and velocity can be perturbed in these directions and the resulting position and velocity still satisfy, to first-order, all the terminal state constraints. This means the costate vector at the final time must be orthogonal to the 6-vector composed of the velocity and gravity vectors. However, note that H_{NT} is just the

inner product of the latter vector with the costate vector; hence $H_{NT}(t_f)$ must be equal to zero, as verified by Figure 6-8.

Alternatively, note that a circular orbit can be fully described via semi-major axis, eccentricity, inclination and node, all of which are constants in the absence of thrust; hence, $H_{NT}(t_f)$ will automatically be zero [46] if the trajectory in question is in fact the desired optimal trajectory.

The multiple shooting algorithm as implemented here converges reliably and efficiently once the guessed parameters are in the neighborhood of the solution. In fact, the convergence rate is quadratic, both theoretically (i.e., Newton's method) and experimentally. The quadratic convergence is desirable from a guidance standpoint because once a converged solution is obtained, the algorithm would very quickly home in on the solution on subsequent guidance cycles. This property indicates that closed-loop guidance can be implemented via cyclical re-optimization. This is done in the next chapter. From an initial guess generated as described above, approximately 40 iterations are required. This aspect of the algorithm could use improvement. Such behavior indicates that the region of convergence is not sufficiently large. The computational burden, even with a few dozen iterations, is probably manageable with today's flight computers. It was also observed that when the destination orbit was raised to higher altitudes, convergence was more reliable. This would suggest that cases where there is a clear distinction between the single burn solution and the multi-are solution converge better than cases where the two solutions are more similar. The possibility of non-convergence and lack of a convergence proof (mathematical or heuristic) poses a barrier to acceptance as a real-time method. This is the case with other numerical methods that have been proposed as online algorithms in recent years [25,41]. Characteristic of these methods is multi-dimensional root isolation. A method for computing abort trajectories that avoids multi-dimensional root-finding is described in Chapter 8.

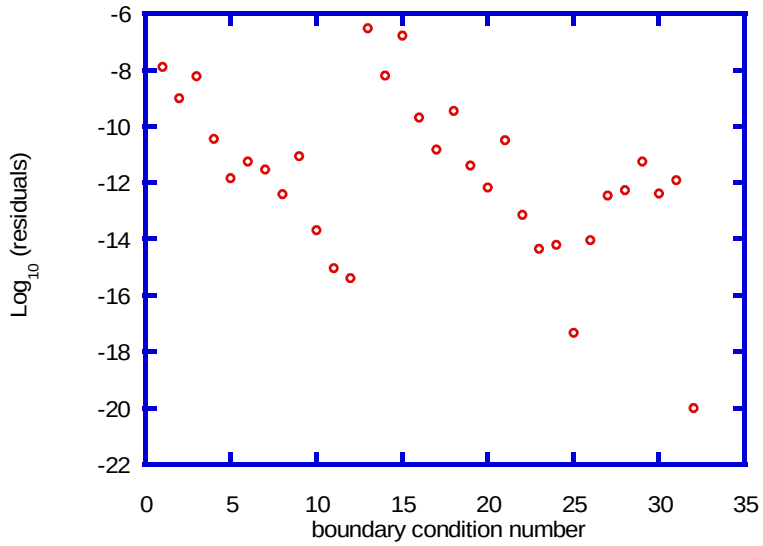

Figure 6-1: Base-10 Logarithm of the Absolute Value of Boundary Condition Residuals

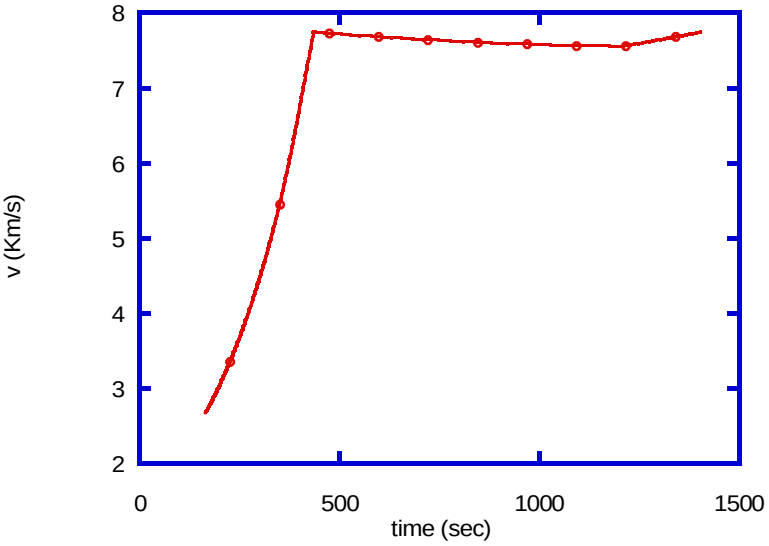

Figure 6-2: Speed vs Time on Optimized Burn-Coast-Burn Trajectory

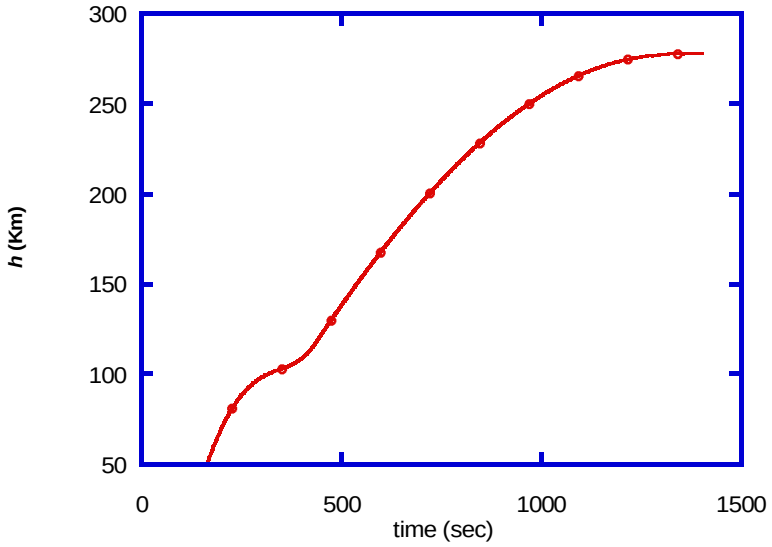

Figure 6-3: Altitude vs Time on Optimized Burn-Coast-Burn Trajectory

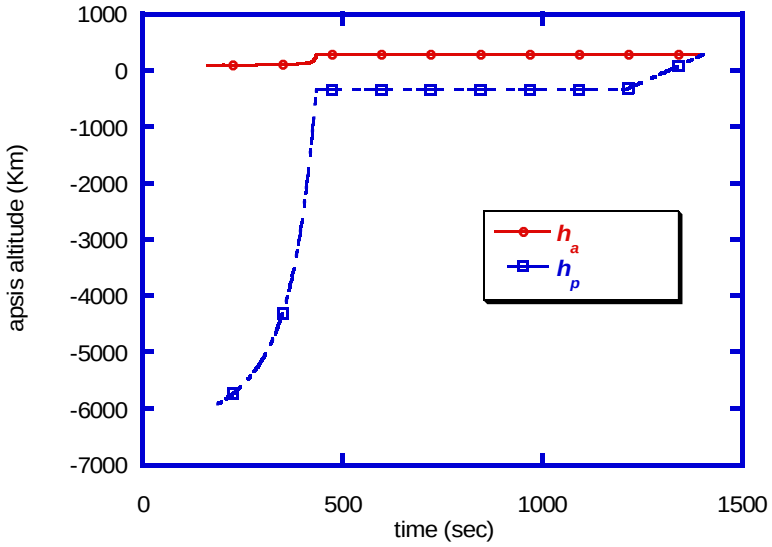

Figure 6-4: Apogee/Perigee vs Time on Optimized Burn-Coast-Burn Trajectory

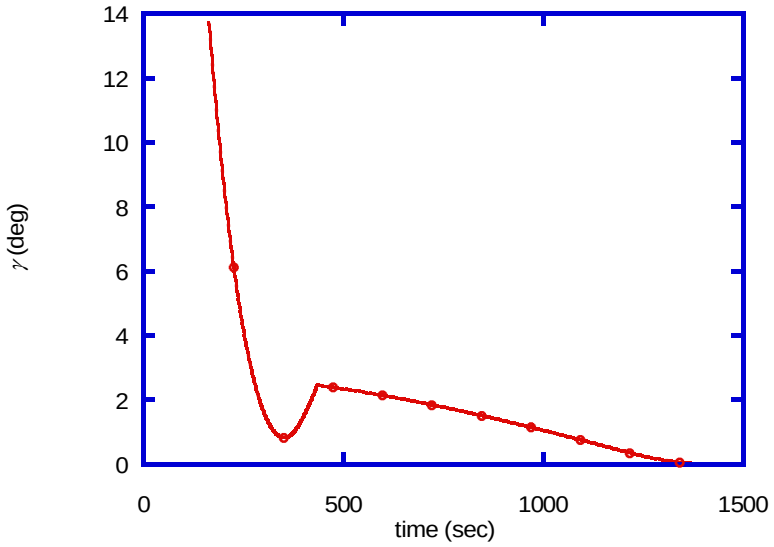

Figure 6-5: Flight Path Angle vs Time on Optimized Burn-Coast-Burn Trajectory

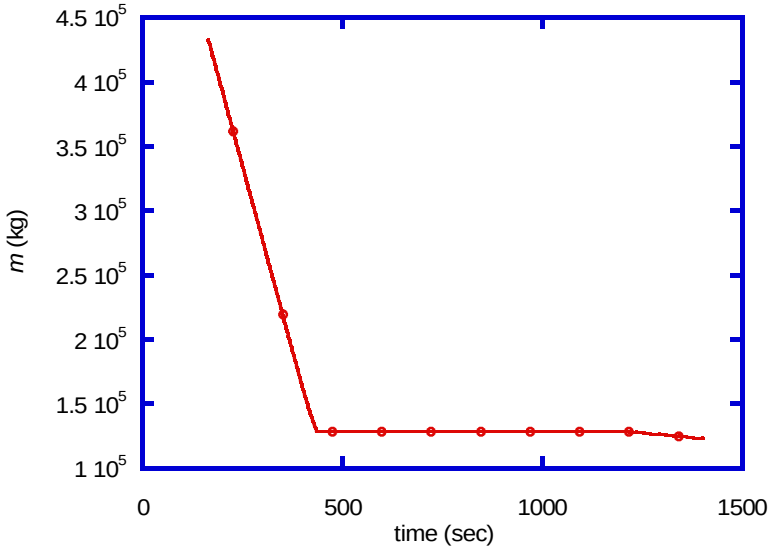

Figure 6-6: Vehicle Mass vs Time on Optimized Burn-Coast-Burn Trajectory

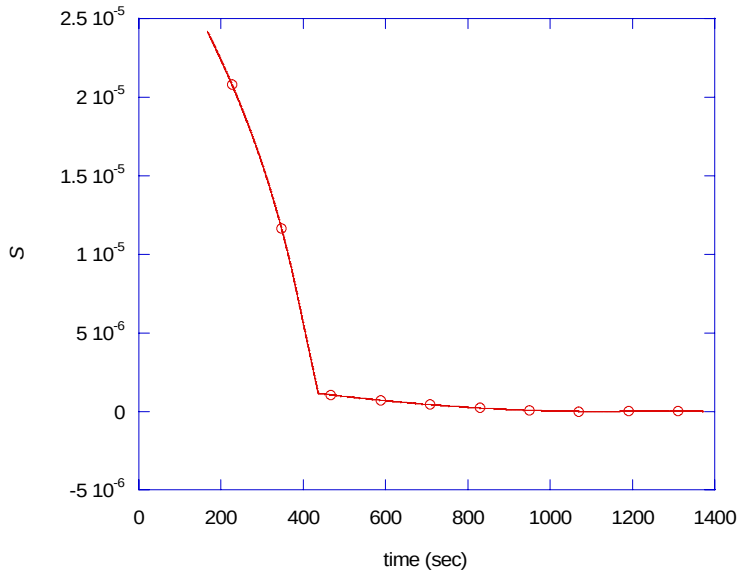

Figure 6-7: Switching Function vs Time on Optimized Burn-Coast-Burn Trajectory

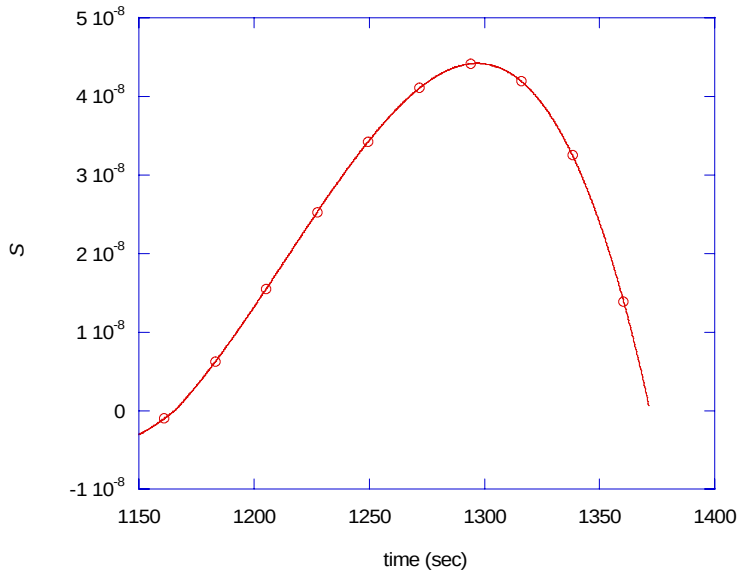

Figure 6-8: Switching Function vs Time on Optimized Burn-Coast-Burn Trajectory

CHAPTER 7

MULTI-ARC CLOSED LOOP GUIDANCE

The previous chapter presented the theoretical development and open-loop, optimized, burn-coast-burn trajectories for a two-stage-to-orbit vehicle. Gath and Calise [25] computed open-loop, burn-coast-burn trajectories for a single-stage-to-orbit vehicle. The present chapter extends previous work by developing a closed-loop guidance protocol for multi-arc ascent to orbit. It was found that, in the neighborhood of the solution, convergence is quick and reliable and, moreover, each Newton iteration requires but a small fraction of a second. This suggests that closed-loop guidance can be implemented by cyclically re-optimizing the trajectory. This is the adopted protocol discussed in the sequel. An alternative protocol, not pursued here, is to take the optimized first burn end conditions (radius, speed, flight path, inclination, node) from an open-loop multi-arc solution, and use them as target conditions in a single-burn guidance protocol. This is effectively what the space shuttle onboard guidance does except the optimized first burn target conditions come from pre-mission analysis. The advantage of cyclically re-optimizing the entire burn-coast-burn sequence is additional adaptivity to off-nominal conditions, i.e., the optimal first burn end conditions change as the vehicle flies and encounters off-nominal dispersions.

7.1 Closed-Loop Guidance Logic for Burn-Coast-Burn Ascent

Several extensions to the existing guidance algorithm were necessary to make it capable of performing closed-loop guidance for burn-coast-burn ascent wherein the entire ascent trajectory (from the current navigated state to end of the OMS burn) is re-optimized every (1 Hz) guidance cycle. The guidance is designed to issue engine cutoff commands, t_{co}, that effectively turn off the main engines at the MECO event and the OMS engines upon circularization. The (predicted) orbital energy at MECO and at OMS cutoff, E_p, extracted from the boundary value solution, is used to calculate the time at which the engines should be shut down. The time rate of change of orbital energy, E, can be estimated by:

$$\dot{E} = \mathbf{a}_{nav}^{T} \mathbf{v}_{nav} \tag{7-1}$$

where $\mathbf{a}_{\text{nav}}$ is the contact (AKA sensed, i.e., doesn't include gravity terms) acceleration vector from the navigation subsystem and $\mathbf{v}_{nav}$ is the navigated inertial velocity vector. Note that when gravity is the only acting force, then energy rate is zero as it should be and similarly, when the net contact force has a component along the velocity vector, energy increases, as it should. The commanded engine cutoff time, t_{co}, then, is given by:

$$t_{co} = t_{nav} + \frac{E_p - E_{nav}}{\dot{E}} \tag{7-2}$$

where t_{nav} is the time of the current guidance cycle, and E_{nav} is the vehicle's current energy given by

$$E_{nav} = \frac{\mathbf{v}_{nav}^T \mathbf{v}_{nav}}{2} - \frac{\mu}{|\mathbf{r}_{nav}|} \tag{7-3}$$

The engine cutoff time commands could alternatively be based on the corresponding times from the boundary value problem solution but the procedure described above is more robust (i.e., less sensitive to mis-modeling), partly because the boundary value problem is solved each guidance cycle only up until several seconds before engine cutoff is about to occur. Beyond this point, the vehicle is flown 'open-loop' (to avoid classical ill-conditioning of the trajectory optimization problem as time-to-go becomes small) until engine cutoff.

The guidance also issues an engine ignition time command, t_{ig}, to initiate the OMS burn. This time is readily available from the boundary value problem solution. Finally, each guidance cycle, the current time is compared to the time corresponding to the first interior node. When the current time is greater than the time of the first interior node, that node and all data associated with the node (e.g., state/costate guess, free or fixed, etc.) are eliminated from the boundary value problem.

7.2 Examination of Various Terminal State Constraints

During the first (main engines) burn, the trajectory optimization problem solved by the guidance consists of determining the thrust direction history and OMS burn start/stop times such that fuel usage is minimized and the vehicle inserts into a circular orbit of specified radius, r_t, and orbit plane specified by a node, Ω_t, and inclination, I_t. The natural thing to do post-MECO is to simply continue with the same set of terminal state constraints and treat the guidance problem as a coast-burn trajectory optimization problem. Another possibility is to re-formulate the post-MECO guidance problem as a coast-burn-coast-burn sequence, which in general will result in

better fuel usage than the coast-burn sequence (using the same terminal state constraints). This guidance option is fairly complicated relative to the expected benefit.

A simpler alternative is to sub-divide the post-MECO circularization problem into a sequence of two coast-burn optimization problems, the first of which is the post-MECO coast to apogee where a perigee-raise burn is performed. Once the perigee-raise burn is performed, the vehicle coasts to the next apsis where a circularization burn is performed to complete the orbital insertion. Whether or not the second OMS burn is actually needed, or executed (i.e., after the perigee-raise burn, the vehicle's orbit parameters may or may not be sufficiently close to the desired parameters), depends on such things as orbit accuracy required (i.e., how far off from the reference circular orbit and reference plane can the final orbit be?), effects of dispersions such as mis-modeled drag, oblate gravity effects, uncertain propulsion parameters, and operational constraints.

The particular terminal state constraints to be prescribed in the coast-(perigee-raise) burn problem then should be based on the aforementioned considerations. Presumably, intelligent decision-making logic, at a higher level than guidance, would dictate to guidance which terminal state constraints to use. This will be made more clear in the sequel where five different terminal state constraint sets are examined that are applicable to the coast-perigee-raise burn optimization problem and each of which has potential benefits that can be exploited for certain vehicles and/or missions. In particular, we'll evaluate the effects on fuel usage, insertion accuracy, and attitude maneuver requirements.

The five constraint sets studied are expressed mathematically as:

$$\left.\begin{aligned} |\mathbf{r}| - r_t &= 0 \\ |\mathbf{v}| - \sqrt{\frac{\mu}{r_t}} &= 0 \\ \mathbf{r}^T\mathbf{v} &= 0 \\ \mathbf{r}^T\mathbf{h}_t &= 0 \\ \mathbf{v}^T\mathbf{h}_t &= 0 \end{aligned}\right\} \tag{7-4}$$

$$\left.\begin{aligned} |\mathbf{v}| - \sqrt{\frac{\mu}{r}} &= 0 \\ \mathbf{r}^T\mathbf{v} &= 0 \\ \mathbf{v}^T\mathbf{h}_t &= 0 \end{aligned}\right\} \tag{7-5}$$

$$\left.\begin{aligned} |\mathbf{r}| - r_t &= 0 \\ |\mathbf{v}| - \sqrt{\frac{\mu}{r_t}} &= 0 \\ \mathbf{r}^T\mathbf{v} &= 0 \\ \mathbf{v}^T\mathbf{h}_t &= 0 \end{aligned}\right\} \tag{7-6}$$

$$\left.\begin{aligned} &\frac{\mathbf{v}^T\mathbf{v}}{2}-\frac{\mu}{r}-\left(-\frac{\mu}{r+r_t}\right)=0 \\ &\mathbf{r}^T\mathbf{v}=0 \\ &\mathbf{r}^T\mathbf{h}_t=0 \\ &\mathbf{v}^T\mathbf{h}_t=0 \end{aligned}\right\} \tag{7-7}$$

$$\left.\begin{aligned} &\frac{\mathbf{v}^T\mathbf{v}}{2}-\frac{\mu}{r}-\left(-\frac{\mu}{r+r_t}\right)=0 \\ &\mathbf{r}^T\mathbf{v}=0 \\ &\mathbf{v}^T\mathbf{h}_t=0 \end{aligned}\right\} \tag{7-8}$$

where $\mathbf{h}_t$ is the target unit normal vector constructed from the target inclination, i_t, and target node Ω_t:

$$\mathbf{h}_t=\begin{Bmatrix}\sin(i_t)\sin(\Omega_t)\\ -\sin(i_t)\cos(\Omega_t)\\ \cos(i_t)\end{Bmatrix} \tag{7-9}$$

and r_t is the circular target radius magnitude. The third constraint of (7-4) ensures that the flight path angle is zero, while the third and fourth constraints of (7-4) taken together enforce the prescribed inclination and node targets (note these could equivalently be replaced with (the more complicated) expressions for inclination and node). Constraint set (7-4) strictly enforces, at the end of the burn, a circular orbit of specified radius in a specified orbit plane. Constraint sets (7-5) through (7-8) represent various 'relaxed' (relative to (7-4)) constraint sets.

Constraint set (7-5) results in the vehicle circularizing at an optimum altitude with relaxed orbit plane constraints. The first of (7-5) simply forces the final orbit to be circular while the third constraint of (7-5) nulls the vehicle's velocity component along the target normal. The latter constraint results in the vehicle's maximum relative latitude at the end of the burn being approximately equal to the relative latitude that the vehicle has when the burn takes place. Thus, the maximum relative latitude is reduced. From spherical trigonometry, the maximum relative latitude is equal to the relative inclination; hence, the relative inclination is reduced. The relative inclination, that is, the angle between the vehicle's angular momentum vector and the target angular momentum vector, is a metric for measuring the amount of plane error. The fuel required for attaining the target inclination and node is directly proportional to the relative inclination. The terminology, relative latitude, ϕ_{rel}, and relative inclination, i_{rel}, becomes clear if the target orbit plane is thought of as being equal to the equatorial plane, e.g., note the formulas:

$$\begin{aligned} \phi_{rel} &= \sin^{-1}\left(\hat{\mathbf{r}}^T \mathbf{h}_t\right) \\ i_{rel} &= \cos^{-1}\left(\hat{\mathbf{h}}^T \mathbf{h}_t\right) \end{aligned} \tag{7-10}$$

which are easily recognized as conventional latitude and inclination formulas, resp., when the target normal, $\mathbf{h}_t$, is replaced with the Earth polar axis. In any case, this constraint is useful, for example, when multiple burns are scheduled (e.g., a rendezvous and docking mission) so that the plane error can be gradually, and efficiently, reduced. Another advantage of this constraint is that it is a function of velocity components only, which generally results in smaller attitude maneuvers and better guidance behavior compared to terminal constraints that are functions of position components. Constraint set (7-5) is useful when the tolerances on orbital altitude and orbital plane are relatively large.

Constraint set (7-6) simply relaxes the orbital plane constraints compared to constraint set (7-4). The first of constraint set (7-7), when combined with the second of (7-7) results in the 'next' apsis radius matching the target radius magnitude. An advantage of constraint set (7-7) over constraint set (7-4) is that the former relaxes the hard constraint on position magnitude. Note that there is no ambiguity in the first of (7-4) regarding whether the next apsis is, or should be, an apogee or perigee. Once the perigee-raise burn is done, the vehicle coasts approximately 180 degrees to the (newly established) next apsis and does a final circularization burn, if desired. The second constraint of (7-7) is justified, in addition to its necessity for the first of (7-7) to make sense, by the fact that apsis modification burns are typically made when the vehicle is very near an apsis (in fact, if the impulsive approximation is invoked, the optimal burn placement is at the apsis.)

Constraint set (7-8) is the same as (7-7) except the orbit plane constraints are relaxed. Five simulation runs using the Maveric simulation with closed-loop guidance were made. The guidance was run at 1 Hz from second stage initiation (at approximately 165 s) through the end of the perigee-raise burn. Up until 1,000 s (about 200 s before the start of the perigee-raise burn), constraint set (7-4) is active in the guidance. At 1,000 s, the active constraint set in the guidance is changed from set (7-4) to a pre-selected constraint set (7-5) – (7-8). The transversality conditions of optimality are handled numerically [47] so that no effort was expended to analytically determine transversality condition expressions. Table 7-1 summarizes results at the end of the perigee-raise burn for each of the five constraint sets.

Table 7-1: Results of Perigee-Raise Burn Using Various Constraint Sets

Terminal State Constraint Set	Delta apogee, n.m. (150 ref.)	Delta perigee, n.m. (150 ref.)	Delta inclination, deg (28.5 ref.)	Delta node, deg (270 ref.)	OMS burn time, s
1 Circularize at specified radius and orbit plane	0.183	-0.0018	-1.1e-4	-1.0e-4	184.675
2 Circularize at optimum radius and null out-of-plane velocity	-1.4439	-1.4748	-0.0056	-0.0706	164.415
3 Circularize at specified radius, null out-of-plane velocity	0.1505	-0.0039	-0.0054	-0.071	172.565
4 Constrain next apsis and orbit plane	-0.003	-2.3737	-2.0e-5	-1.0e-4	176.615
5 Constrain next apsis, null out-of-plane velocity	0.0107	-1.4037	-0.0055	-0.0706	165.145

As expected, the relaxed constraint sets (7-5) – (7-8) all have better fuel efficiency than the nominal constraint set (7-4). Constraint set (7-5), circularize at optimum altitude with relaxed

plane constraints, had the best fuel performance (approximately 11 percent better than with set (7-4)), albeit with noticeably larger errors (1.5 n.m. apses errors compared to 0.183 n.m., and 0.006 deg inclination error compared with 1.1e-4 deg) than that of set (7-4). However, there are likely many mission scenarios where these errors would be tolerable and the fuel savings would be beneficial.

Figure 7-1 shows the number of Newton iterations per guidance cycle (Note: the legend is in order of constraint sets) vs. time. Apparently, it generally takes about 12 iterations for guidance re-convergence at the 1,000 second mark when a different constraint set is activated in the guidance. It takes about five iterations per guidance cycle during nominal operation. Thanks to modern computing power, even 12 iterations require only a small fraction of a second. Note that during the OMS burn itself, only one iteration per guidance cycle is required. During the OMS burn, single shooting is used as compared with multiple shooting prior to the OMS burn. Apparently, scaling, or lack thereof, of the continuity conditions needs to be revisited. From past experience, under nominal operation, i.e., small perturbations, one Newton iteration per guidance cycle should be sufficient to maintain a good guidance solution.

Figures 7-2 and 7-3 show guided simulation results for angles of attack and sideslip histories during the OMS burn for the different constraint sets. As expected, the relaxed constraint sets have relatively modest attitude maneuvering requirements. Constraint sets (7-5), (7-7), and(7-8) which have relaxed ‘in-plane’ constraints result in smooth, low rate angle of attack histories compared to the other two sets. Constraint sets (7-5), (7-6), and (7-8) which have relaxed orbit plane constraints result in near constant angle of sideslip histories in contrast with the other two which include large yaw maneuvers of 50 deg or more. Figure 7-4 shows the angular velocity magnitude histories during the OMS burn. Interestingly, constraint set (7-5) gives the best translational fuel performance as well as the best attitude profile (smallest amount of attitude maneuvering) and set (7-8) provides the second best translational performance and the second

best rotational performance. Set (7-5) results in rotational rates and total maneuver magnitude of roughly an order of magnitude better than set (7-4). Small commanded rates are especially desirable for large vehicles or for vehicles that don't have a lot of attitude control authority. In defense of set (7-4), the accuracy attainable means it would almost certainly be unnecessary to do another OMS burn, which could be operationally advantageous.

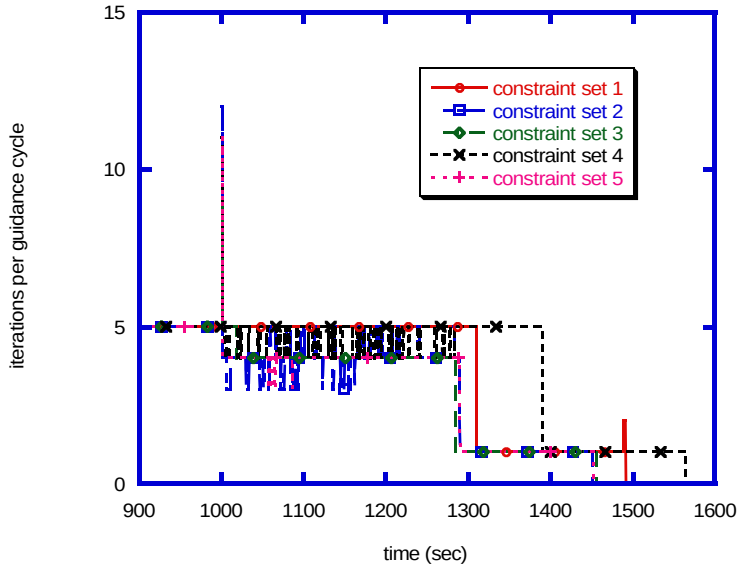

Figure 7-1: Number of Iterations Per Guidance Cycle vs Time

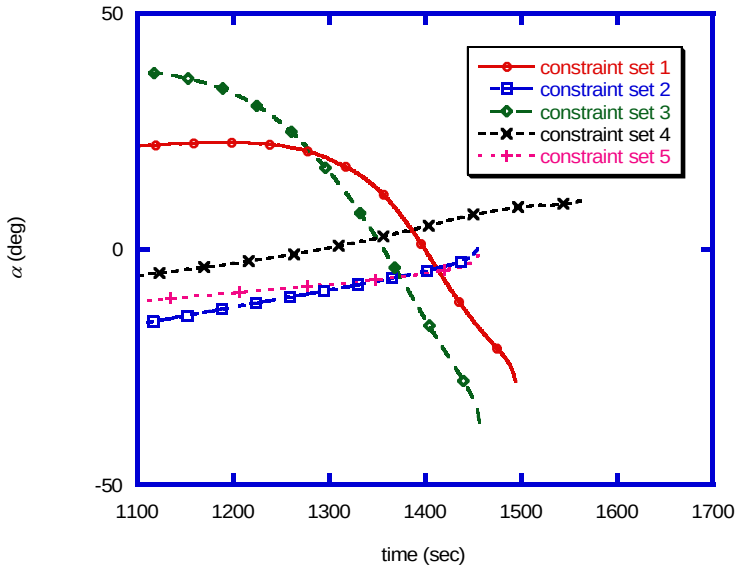

Figure 7-2: Angle of Attack vs Time During Perigee-Raise Burn

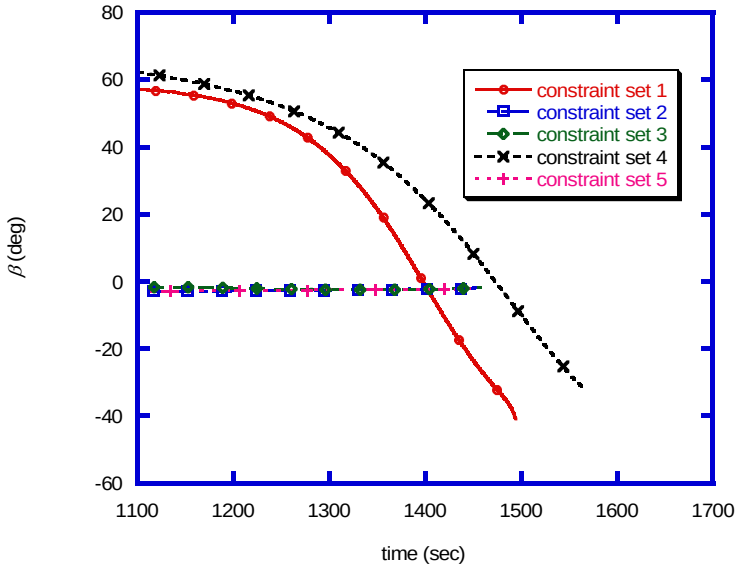

Figure 7-3: Angle of Sideslip vs Time During Perigee-Raise Burn

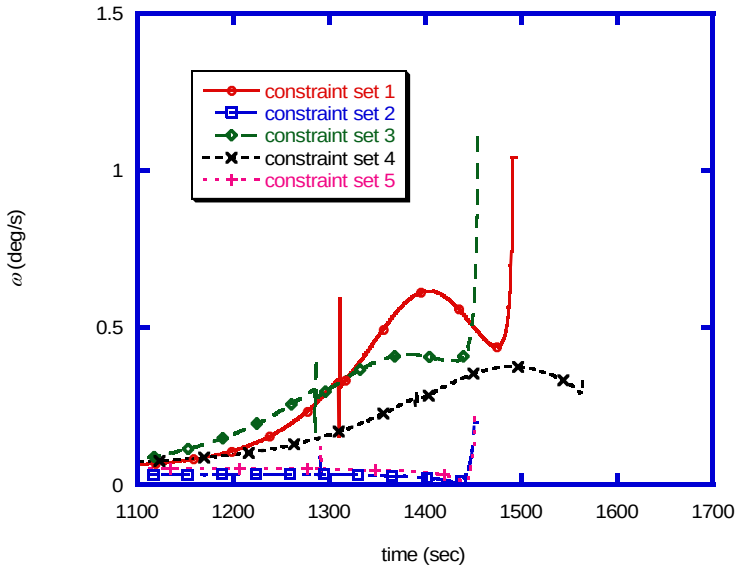

Figure 7-4: Vehicle Angular Velocity Magnitude During Perigee-Raise Burn

CHAPTER 8

ABORT TRAJECTORY GENERATION

Mission aborts during the ascent trajectory of a reusable launch vehicle can occur for several reasons. The vehicle's integrated health management system may detect a problem in the life support system, one or more engines may unexpectedly shut down, or a piece of debris may strike the vehicle in a critical area making it unsafe to continue the mission. Hence, abort scenarios potentially could involve flight with either full propulsion capability (i.e., life support system example) or, flight with reduced propulsion capability (i.e., one or more engines out). There are two major classes of abort, intact and contingency. Intact aborts are those which allow for safe recovery of the crew and vehicle. Possible intact aborts consist of return-to-launch site (RTLS), abort to downrange landing site (ATDS, also known as trans-Atlantic abort landing in space shuttle parlance), abort to orbit (ATO), and abort once around (AOA) [3]. Contingency aborts are those aborts wherein significant propulsion capability is lost making safe recovery of the vehicle physically impossible. This chapter focuses on the RTLS and ATDS intact aborts. Contingency abort guidance typically needs to be handled on a vehicle-specific basis as it will involve many vehicle-specific flight rules and constraints, particularly for manned vehicles. ATO and AOA guidance can typically be handled, without too much modification, by nominal powered guidance algorithms for, by implication, the vehicle's propulsion capability is not significantly reduced and/or the abort occurs late enough in the ascent that steering the rest of the way to orbit is not difficult. Calise and Brandt [28] treat the AOA case using a hybrid method.

The abort trajectory problem can be treated using the techniques of Chapters 2 and 3. The principal difference, aside from possibly degraded vehicle performance, is in the terminal state constraints. In the case of abort to downrange or abort to launch site, the target values are derived from an entry profile [48] that is pre-defined to provide acceptable entry conditions compatible with a successful entry and landing. The space shuttle guidance [3] involves a strategy wherein a linear relationship between speed and range to the landing site defines the cutoff targets for ascent guidance along with zero flight path angle and heading angle aligned with the landing site. The effects of atmospheric forces are ignored. Calise and Brandt [28] use an approach reminiscent of that of Chapter 2 of this thesis in generating burn-coast abort trajectories. In that paper, it is suggested that throttling may provide advantages for abort trajectory generation. They also report convergence problems in isolating the multi-dimensional roots required in their formulation. The present formulation addresses both the throttling and the non-convergence concerns.

In this chapter, a simplified procedure for generating powered abort to launch/landing site trajectories is presented. The procedure reduces the associated nonlinear two-point boundary-value problem to that of a one-dimensional minimization of a smooth uni-modal function. The generated trajectory extends from the time of abort initiation, at some point early in the ascent, to the point of fuel depletion where it is required that the vehicle's position and velocity be compatible with a gliding reentry, i.e., appropriate altitude and speed versus range-to-landing site at burnout, with vehicle velocity directed to the targeted launch or landing site. Upon completion of the powered phase, it is assumed that the entry and landing guidance system is used to steer the vehicle to the runway. Closed-loop powered flight guidance is effected by either cyclically recomputing the trajectory each guidance cycle or, by re-computing the trajectory only on an as-needed basis and using a feedback control law to track the reference trajectory in-between updates. Guidance is discussed in greater detail in the next chapter.

The abort trajectory generation technique developed here could be used in pre-mission studies to quickly assess the characteristics of fuel-efficient RTLS and ATDS trajectories for a given vehicle. It could also be used as part of an onboard real-time trajectory generation and guidance scheme. The analysis to date on the present techniques has been done assuming that the vehicle is in a flight regime where aerodynamic forces are a small fraction of propulsive forces. For the case of early intact abort (i.e., retention of a sufficient part of the vehicle's propulsion capability), the guidance techniques of Chapters 2 and 3 could be used (using, say, nominal orbit main engine cutoff targets) to steer the vehicle through the atmosphere whereupon the protocol developed in this and the next chapter could be used for subsequent guidance and steering.

8.1 General Development

Start off by assuming that the vehicle's position vector components at any point along the generated trajectory can be expressed as a cubic polynomial in time, t, referenced to some reference time, t_0.

$$\mathbf{r}(t) = \mathbf{r_0} + \mathbf{v_0}t + \mathbf{a_0}\frac{t^2}{2} + \mathbf{j_0}\frac{t^3}{6} \tag{8-1}$$

where $\mathbf{r}_0$ is the vehicle's initial position vector expressed in an Earth-centered Earth-fixed frame, $\mathbf{v}_0$ is the vehicle's initial Earth-relative velocity (i.e., the velocity as seen by an observer fixed in the rotating Earth) vector, $\mathbf{a}_0$ is the initial total acceleration vector acting on the vehicle as observed in the rotating Earth frame, and $\mathbf{j}_0$, also known as jerk [49], is the time rate of change of $\mathbf{a}_0$. Given (8-1), the vehicle's relative velocity and acceleration at any time are apparently quadratic and linear expressions, respectively:

$$\begin{aligned}\mathbf{v}(t) &= \mathbf{v}_0 + \mathbf{a}_0 t + \mathbf{j}_0 \frac{t^2}{2} \\ \mathbf{a}(t) &= \mathbf{a}_0 + \mathbf{j}_0 t\end{aligned} \tag{8-2}$$

For a specified time of powered flight, T, the final position and velocity, $\mathbf{r}_f$ and $\mathbf{v}_f$, are given by:

$$\begin{aligned}\mathbf{r}_f &= \mathbf{r}(T) = \mathbf{r}_0 + \mathbf{v}_0 T + \mathbf{a}_0 \frac{T^2}{2} + \mathbf{j}_0 \frac{T^3}{6} \\ \mathbf{v}_f &= \mathbf{v}(T) = \mathbf{v}_0 + \mathbf{a}_0 T + \mathbf{j}_0 \frac{T^2}{2}\end{aligned} \tag{8-3}$$

If it is assumed that the final position and velocity components are available, perhaps computed using targeting logic to be described in the sequel for example, then equations (8-3) comprise 6 linear equations in the 6 unknown components of $\mathbf{a}_0$ and $\mathbf{j}_0$. The solution is given by:

$$\begin{aligned}\mathbf{a}_0 &= -\frac{6(\mathbf{r}_0 - \mathbf{r}_f) + 2T(2\mathbf{v}_0 + \mathbf{v}_f)}{T^2} \\ \mathbf{j}_0 &= \frac{12(\mathbf{r}_0 - \mathbf{r}_f) + 6T(\mathbf{v}_0 + \mathbf{v}_f)}{T^3}\end{aligned} \tag{8-4}$$

Using acceleration and jerk from (8-4) in (8-1) and the first of (8-2), the resulting polynomial expressions for position and velocity identically satisfy the boundary conditions:

$$\begin{aligned}\mathbf{r}(t=0) &= \mathbf{r}_0 \qquad & \mathbf{v}(t=0) &= \mathbf{v}_0 \\ \mathbf{r}(t=T) &= \mathbf{r}_f \qquad & \mathbf{v}(t=T) &= \mathbf{v}_f\end{aligned} \tag{8-5}$$

The next section describes how final position and velocity are obtained for the abort trajectory generation application. These vectors will represent the target values which must be attained upon termination of the powered flight maneuver such that the entry and landing guidance system can steer the vehicle safely to the landing site.

8.1.1 Construction of Target Position and Velocity Vectors

Given initial position and velocity, a specified final range-to-go, time of flight, *T,* and landing site coordinates, construct the target position and velocity vectors as follows. First, define a convenient reference frame, hereinafter referred to as the guidance frame, where the *x*-axis is directed from the center of the Earth through the desired landing site (near the launch site in the case of RTLS abort or some down range landing site that lies near the nominal ground track in the case of ATDS.) The *z*-axis is defined such that the *x*-*z* plane contains the center of the Earth, the landing site and the initial vehicle position and the *z*-axis is directed from the landing site to the vehicle. The y-axis completes the orthogonal right-handed set. Note that the guidance frame is fixed in the rotating Earth and hence, is non-inertial The unit vectors along the *x*-, *y*- and *z*-coordinate axes are, in terms of the landing site vector, $\mathbf{r}_L$, and the initial position vector, $\mathbf{r}_0$:

$$\hat{\mathbf{x}} = \frac{\mathbf{r}_L}{|\mathbf{r}_L|} \qquad \hat{\mathbf{y}} = \frac{\mathbf{r}_0 \times \mathbf{r}_L}{|\mathbf{r}_0 \times \mathbf{r}_L|} \qquad \hat{\mathbf{z}} = \hat{\mathbf{x}} \times \hat{\mathbf{y}} \tag{8-6}$$

where $\mathbf{r}_0$ and $\mathbf{r}_L$ are expressed in the same coordinate axes.

Next assume that there is available a set of acceptable entry altitudes and velocities versus range-to-go to the landing site. This set could be in the form of tabular data, or it could be in functional form such as an approximate linear fit. In either case, the data are derived from an

entry trajectory analysis for the particular vehicle so that appropriate altitude and relative speed for a given range-to-go R can be readily computed:

$$h = h(R) \qquad v = v(R) \tag{8-7}$$

Figures 8-1 and 8-2 show typical entry altitude versus range-to-go and relative velocity versus range-to-go profiles.

Now, given a range-to-go, R, compute an intermediate position vector expressed in the guidance frame:

$$\mathbf{r}_{tgtI} = \left(r_E + h(R)\right)\left[\cos\left(\frac{R}{R_E}\right) \quad 0 \quad \sin\left(\frac{R}{R_E}\right)\right]^T \tag{8-8}$$

where r_E is the radius of the Earth below the vehicle's initial position. Note that this intermediate position vector is in the x-z plane and is R units downrange of the landing site. Next, temporarily leave this immediate topic and address construction of the target velocity vector.

Study of typical entry trajectories indicates that, unlike altitude and speed, flight path angle is a very weak function of range-to-go. In fact, flight path angle histories are typically transitory in nature, therefore it is undesirable to directly obtain target flight path angle from a nominal entry profile. Instead the target flight path angle is set equal to a typical value of, say, negative 1 deg. Although not addressed in the current analysis, flight path angle could conceivably be reserved as another parameter to vary in search of a feasible abort trajectory. Now, given a range-to-go, R, and a target flight path angle, γ_{tgt}, compute an intermediate velocity vector:

$$\mathbf{v}_{tgtI} = v(R)\begin{bmatrix} \cos\left(\frac{R}{R_E}\right)\sin(\gamma_{tgt}) + \sin\left(\frac{R}{R_E}\right)\cos(\gamma_{tgt}) \\ 0 \\ \sin\left(\frac{R}{R_E}\right)\sin(\gamma_{tgt}) - \cos\left(\frac{R}{R_E}\right)\cos(\gamma_{tgt}) \end{bmatrix} \tag{8-9}$$

Note that this velocity vector is also in the *x*-*z* plane. Thus, (8-8) and (8-9) together imply that the vehicle burns out in the *x*-*z* plane with the velocity vector contained in the *x*-*z* plane. For cases where the vehicle has a significant initial out-of-plane velocity component, use of (8-8) and (8-9) would result in thrust unnecessarily being applied to counter the out of plane motion. Dog leg maneuvers during nominal ascent and abort landing sites displaced from the nominal ground track are examples where this would be the case. In order to reduce this effect, rotate the target position and velocity about the guidance frame *x*-axis through a small angle that represents the approximate out-of-plane distance through which the vehicle would naturally drift over the powered burn.

$$\mathbf{r}_{tgt} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(-\psi) & \sin(-\psi) \\ 0 & -\sin(-\psi) & \cos(-\psi) \end{bmatrix}\mathbf{r}_{tgtI} \qquad \mathbf{v}_{tgt} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(-\psi) & \sin(-\psi) \\ 0 & -\sin(-\psi) & \cos(-\psi) \end{bmatrix}\mathbf{v}_{tgtI}$$
$$\psi = \tan^{-1}\left(\frac{-\mathbf{v}_{0-y}\,T/2}{\mathbf{r}_{tgtI-z}}\right) \tag{8-10}$$

Note that the final target velocity vector incorporates the desired flight path angle, the desired speed, and points to the landing site. Likewise the final target position vector has the desired altitude, specified range from the landing site and is contained in a plane near which the vehicle will naturally drift over the course of the powered burn.

The next section discusses how to calculate the propulsive ΔV required to fly a trajectory characterized by (8-1), for a specified final range-to-go, and specified time of flight and given initial position and velocity. This calculation is critical in determining candidate trajectories whose required ΔV match the estimated propulsive ΔV remaining on the vehicle. This ensures that the powered abort trajectory depletes all the fuel.

8.1.2 Determination of ΔV Required

The assumed form for the acceleration (8-2) represents the total acceleration acting on the vehicle as seen in an Earth-fixed frame. The relationship between the relative acceleration, (8-2), and the inertial acceleration, $\mathbf{a}_I$, is given by:

$$\mathbf{a}(t) = \mathbf{a}_I(t) - 2\boldsymbol{\omega} \times \mathbf{v}_r - \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}) \tag{8-11}$$

where $\mathbf{a}_I$ is the sum of the propulsive, aerodynamic and gravity acceleration. The small centrifugal term ($\sim\omega^2 r$) is neglected. It is assumed that the Coriolis term is small (and linear) enough that it can be approximated with an average value over the trajectory, namely

$$\mathbf{C}_{avg} = -\boldsymbol{\omega} \times (\mathbf{v}_0 + \mathbf{v}_f) \tag{8-12}$$

as its contribution to ΔV is assumed to be small compared to gravity and propulsion. Note that this assumption is very good for RTLS trajectories because the vehicle speed stays relatively small. For ATDS trajectories, the assumption degrades somewhat. Even for the ATDS trajectories however, it is expected that the character of the generated trajectories to be the same whether or not the average Coriolis assumption is invoked.

It is assumed that the ΔV contribution due to aerodynamics forces is small enough, relative to gravity and propulsion, to neglect. It is also assumed that the vehicle's spatial variation over the trajectory is small enough that an average gravity field model is sufficient:

$$\mathbf{g}_{avg} = -\frac{\mu}{2}\left(\frac{\mathbf{r}_0}{\left|\mathbf{r}_0\right|^3} + \frac{\mathbf{r}_{tgt}}{\left|\mathbf{r}_{tgt}\right|^3} \right) \tag{8-13}$$

Analogous comments apply to the average gravity assumption as to the small Coriolis assumption. Later, it is mentioned how to relax these assumptions and shown that doing so adds but a small additional computational burden to the trajectory generation procedure.

With the above assumptions and (8-2), it is possible to obtain a simple expression for the propulsive acceleration, $\mathbf{a}_p$:

$$\begin{aligned} &\mathbf{C}_{avg} + \mathbf{g}_{avg} + \mathbf{a}_p(t) = \mathbf{a}(t) = \mathbf{a}_0 + \mathbf{j}_0 t \\ &\Rightarrow \mathbf{a}_p(t) = \mathbf{a}_0 + \mathbf{j}_0 t - \mathbf{g}_{avg} - \mathbf{C}_{avg} \end{aligned} \tag{8-14}$$

Recall that initial acceleration and jerk, $\mathbf{a}_0$ and $\mathbf{j}_0$, respectively, are simple functions (8-4) of the initial and target position and velocity vectors and time of flight T. Finally, the propulsive ΔV required, ΔV_{reqd} is:

$$\Delta V_{reqd} = \int_0^T \left|\mathbf{a}_p(t)\right| dt = I(T) - I(0) \tag{8-15}$$

where the indefinite integral, $I(x)$, is given by:

$$I(x)=\frac{\left|\mathbf{j}_0\right|\left(\mathbf{j}_0^T\mathbf{a}_{p0}+\left|\mathbf{j}_0\right|^2 x\right)\left|\mathbf{a}_p(x)\right|+\left[-\left(\mathbf{j}_0^T\mathbf{a}_{p0}\right)^2+\left|\mathbf{j}_0\right|^2\left|\mathbf{a}_{p0}\right|^2\right]\ln\left[2\left(\frac{\mathbf{j}_0^T\mathbf{a}_{p0}}{\left|\mathbf{j}_0\right|}+\left|\mathbf{j}_0\right|x+\left|\mathbf{a}_p(x)\right|\right)\right]}{2\left|\mathbf{j}_0\right|^3} \tag{8-16}$$

where $\mathbf{a}_{p0}$ is the initial propulsive acceleration. Note that one way to relax the average Coriolis and gravity assumptions is simply to divide the ΔV integral evaluation into two or more pieces, although it is unlikely that such accuracy would ever be needed. Effects of errors in navigation, vehicle mass estimate and propellant remaining would likely dominate deleterious effects caused by gravity and Coriolis modeling errors.

Figure 8-3 shows a typical plot of required propulsive ΔV, for an RTLS scenario, versus final range-to-go using the formulas of this section for a fixed time of flight, T, of 280 s, and "initial" position and velocity taken from a typical nominal ascent at 200 s after liftoff. Note that ΔV is a smooth function of range-to-go and has a well-defined minimum. This implies that it should be easy to determine when the required ΔV exceeds the remaining ΔV for a given value of T. To the right of the minimum, as range-go-go increases, ΔV increases sharply because large range-to-go corresponds to large final altitude and speed. To the left of the minimum, as range-to-go decreases, ΔV sharply increases because the vehicle has to turn around and back-track up range and then do an expensive propulsive deceleration maneuver to end up with a low altitude and speed appropriate for close proximity to the landing site. The minimum corresponds to a range-to-go that neither forces the vehicle to go too far up range nor too far down range. For the case of down range abort, the characteristics for the curves are analogous. Specifying either relatively large or relatively small values of range-to-go is non-optimum, requiring expensive propulsive maneuvers.

The estimated remaining propulsive ΔV on the vehicle, ΔV_{rem}, can be calculated given an estimate of the current vehicle mass, m_0, the empty vehicle mass, m_f, and the average I_{sp} of the propellant:

$$\Delta V_{rem} = -g_0 I_{sp} \ln\left(\frac{m_f}{m_0}\right) \tag{8-17}$$

8.1.3 Feasible Trajectory Search

By varying T, a family of ΔV vs. range-to-go curves, each of which is smooth with a well-defined minimum can be generated. For a given value of T, an efficient one-dimensional minimization algorithm, e.g., Brent's method [38], is used to determine the range-to-go, R_{min}, that yields the minimum ΔV required, ΔV_{min}. If $\Delta V_{min} < \Delta V_{rem,}$ then, e.g., Brent's one-dimensional root-finder [38] is used to determine the value of range-to-go that satisfies the ΔV constraint, that is, $\Delta V = \Delta V_{rem}$. An important point here is that there are, for each value of T, actually two values of range-to-go (one on each side of R_{min}) that satisfy the ΔV constraint. The search is limited to the "large" values of range-to-go under the rationale that large range-to-go implies high altitude and, hence, low dynamic pressure and benign vehicle loads. In the sequel, the trajectory associated with such a solution is referred to as a candidate trajectory. Each value of T will, in general, have a candidate solution to which it is associated, numerically computed as described above. Now, associated with a candidate trajectory, there is a corresponding thrust magnitude profile, $F(t)$, given by:

$$F(t) = m(t)\left|\mathbf{a}_p(t)\right| = m_0 \exp\left(-\frac{\Delta V(t)}{g_0 I_{sp}}\right)\left|\mathbf{a}_p(t)\right| = m_0 \exp\left(-\frac{I(t) - I(0)}{g_0 I_{sp}}\right)\left|\mathbf{a}_p(t)\right| \tag{8-18}$$

where again $I(\,)$ is the indefinite integral (8-16).

Note that the basic premise of our trajectory generation approach provides no explicit control over the thrust profile. One could argue that this is a tradeoff for the well-behaved nature of the functions associated with this method. For a given value of T, there is no guarantee that the candidate solution thus determined will have a physically realizable thrust profile. The thrust could be quite large at some time points and very small at others. Too small a thrust is undesirable because this indicates that the engines should be temporarily deeply throttled or shut down. It is usually undesirable to temporarily shut down an engine, and most rocket engines have limited throttle range. Calling for too large a thrust would of course exceed the vehicle's maximum available thrust.

It can be predicted that, for decreasingly small values of T, the solution is going to look like the classical minimum time solution with unconstrained thrust magnitude, i.e. the bang-bang solution, characterized by large thrust magnitudes. On the other hand, for increasingly large values of T, the solution will be characterized by the classical bang-off-bang, or, specialized to our rocket problem, the burn-coast-burn solution. Somewhere in between small and large values of T, it is expect that solutions with thrust profiles that are realizable can be found.

How is a search for such a thrust profile conducted and what figure of merit can be used that might indicate a realizable thrust profile? First of all, the range of values of T to search over can be substantially reduced if the time, T_{min}, required to burn the (estimated) remaining propellant, assuming constant maximum thrust, is computed. Likewise, the time, T_{max}, required to burn the remaining propellant, assuming constant minimum thrust, can be computed. The min and max values so computed provide plausible values to which to limit the search for acceptable thrust profiles.

The first figure of merit considered, as an indicator of a desirable thrust profile, is the maximum thrust value associated with a candidate solution. If, over the plausible range of values of *T*, the maximum thrust value associated with each candidate solution is minimized, then such a strategy is expected to also yield minimum thrust values that are not undesirably small, for the ΔV requirement along with a reasonable value of T tends to ensure this property. Mathematically, the optimization is expressed as:

$$\min_{T\in[T_{\min},T_{\max}]}\max_{t\in[0,T]}F(t) \tag{8-19}$$

where it is understood that the thrust profile, *F(t),* corresponds to a candidate solution. The function associated with the maximization above is a uni-modal, smooth function making the maximization part of the problem efficient. The composite optimization problem is solved with a sequence of one-dimensional searches each of which involve well-behaved smooth functions.

A second optimization strategy is to minimize, over all plausible values of *T*, the difference between the maximum thrust and the minimum thrust on each candidate trajectory. The idea behind this strategy is to minimize the amount of throttling needed. Mathematically:

$$\min_{T\in[T_{\min},T_{\max}]}\left(\max_{t\in[0,T]}F(t)-\min_{t\in[0,T]}F(t)\right) \tag{8-20}$$

Analysis to date shows that the resulting optimized trajectories using these objectives are very similar. The min/max thrust strategy is computationally less demanding. A third strategy is to minimize the integral of the square of thrust minus desired thrust:

$$\min_{T\in[T_{\min},T_{\max}]} \int_0^T \frac{\left(F(t)-F_{des}\right)^2}{F_{des}^2 T}\,dt \tag{8-21}$$

This is the figure of merit adopted for all numerical results shown in this chapter and in Chapter 9. Because no analytical solution exists, a 10-point Gauss-Legendre quadrature formula [38] is used to evaluate the integral. The desired thrust, F_{des}, is taken to be the average of the minimum and maximum available thrust. This gives us a good chance of avoiding thrust saturation because the solutions will tend to have thrust histories that stay in the neighborhood of the average available thrust. It was found that all three of the optimization strategies result in fairly similar trajectories with physically realizable thrust profiles. The minimum of the maximum thrust strategy may be useful for vehicles with relatively low thrust. The minimize thrust deviations strategy (8-21) may be the most useful of the three because it is the one that comes closest to allowing the user to specify a particular thrust profile.

The next section presents rationale for presuming a cubic polynomial form for the position history. The choice of such a simple form is advantageous in that it results in smooth, well-behaved functions. It turns out that the cubic form is also a wise choice with respect to fuel optimality and flexibility.

8.1.4 Justification for Cubic Position

Two questions come to mind when considering the presumed form of the position history, (8-1). Does the cubic polynomial form allow for enough flexibility to adequately represent (in a

practical sense) all flyable abort trajectories? What about optimality? It would be nice to know that the generated trajectories are at least quasi-optimal so that fuel is not unnecessarily wasted for scenarios where the limits of vehicle capability are approached. These questions are equivalent to asking if the linear form of the acceleration, which results in the cubic position, is adequate.

Consider the "minimum acceleration" variational problem:

$$\min_{\mathbf{a}(t)} \int_{t_0}^{T} \mathbf{a}^T \mathbf{a} dt \tag{8-22}$$

where **a** is the thrust acceleration vector, subject to the differential equations:

$$\begin{aligned} \dot{\mathbf{r}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= \mathbf{g} + \mathbf{a} \end{aligned} \tag{8-23}$$

where a uniform gravity field, **g**, is assumed and it is presumed that there would be some terminal state constraints imposed. Form the Hamiltonian:

$$H = \boldsymbol{\lambda}_r^T \mathbf{v} + \boldsymbol{\lambda}_v^T \left(\mathbf{g} + \mathbf{a} \right) + \mathbf{a}^T \mathbf{a} \tag{8-24}$$

Note that mass is an ignorable state because it doesn't show up anywhere in any meaningful way. From here, it is easy to verify that the velocity costate vector is linear and, taking the partial of the Hamiltonian with respect to acceleration vector, the optimal acceleration vector is equal to the (negated) velocity costate; hence, the optimal acceleration vector, in a uniform gravity field, is linear just as presumed (implicitly) by using a cubic function for position. The uniform gravity field is a rather gross assumption but numerical experience indicates that even for trajectories that

have a large spatial variation (e.g., 20-30 deg or more of range angle traversal), the numerically computed optimal trajectories are still characterized by very nearly linear steering.

The minimum energy performance index does not yield the same trajectories as the classical minimum fuel performance index, but one would expect "minimum energy" trajectories to be characterized by good fuel efficiency, if not fuel optimality. Having this mathematical justification provides confidence in the potential efficacy of the method.

8.2 Open-Loop Trajectories

A set of open-loop RTLS and ATDS trajectories were generated for the generic two-stage-to-orbit vehicle of Appendix E. The desired thrust value was fixed at 750,000 lb. RTLS abort times of 170, 180, and 190 s, and ATDS abort times of 170, 200, 240, 280, 340, and 430 s were used. The value of 170 s for the first RTLS abort was chosen because before that time dynamic pressure is too high to initiate the required turn around. If an abort were to occur before 170 s, the vehicle would continue flying down range to waste fuel before turning around. For abort times of greater than 190 s, the vehicle has too large a component of velocity in the down range direction to turn around and fly back with the remaining propellant. Nominal MECO occurs at 439 s so the 430 s ATDS case is a difficult case because by that time, the vehicle is almost out of fuel.

Figures 8-4 through 8-10 show the powered portion of the open-loop RTLS trajectories. Figures 8-4 and 8-5 show altitude and speed vs. time, resp. Note that the final altitudes and speeds, and hence final energies, are very nearly the same. The vehicle burns the same amount of total propellant regardless of abort time, so this makes sense. Figure 8-6 shows the ground tracks due east of the Kennedy Space Center launch site. Note that RTLS trajectories are essentially planar trajectories. Figure 8-7 shows the down range vs. time, clearly illustrating the turn around

maneuver. Figure 8-8 shows the mass histories. The final mass was set to 255,000 lbm for all abort cases. Figure 8-9 shows the open-loop thrust histories which correspond to approximately a 45 percent throttle range, which is within the throttling capabilities of many existing and proposed launch vehicles. Figure 8-10 shows the dynamic pressure stays low throughout the maneuver, peaking at the end of the powered maneuver.

Figures 8-11 through 8-17 show the powered portion of the open-loop ATDS trajectories. Dakar, Senegal in western Africa is the specified landing site. Dakar is very nearly on the nominal ground track for a due east launch from Kennedy Space Center. Figures 8-11 and 8-12 show altitude and speed vs. time, resp. As in the case of the RTLS trajectories, the final altitudes and speeds, and hence final energies, are very similar to one another. Figure 8-13 shows the ground tracks which show the vehicle heading to Dakar. Figure 8-14 shows the down range vs. time. The final range-to-go in all cases is between 2,600 and 2,800 n.m. Figure 8-15 shows the mass histories. Figure 8-16 shows the open-loop thrust histories. There is a wide variation of thrust levels in the trajectories. In the next chapter, guided trajectories demonstrate that trajectory control is still acceptable even when the open-loop trajectory calls for more thrust variation than is available. Figure 8-17 shows that the dynamic pressure stays low throughout the maneuver and is more benign than the RTLS trajectories.

All the open-loop trajectories generated here inherently assumed instantaneous attitude maneuvering and throttling. Moreover, the total thrust variations correspond to approximately 80 percent throttling capability. The guided trajectories of the next chapter take into account the fact that the vehicle does not have the correct attitude for the start of the RTLS turn around maneuver. Also, a 50 percent throttling capability is used so that thrust saturation (realistically) occurs for those scenarios where the open-loop trajectory calls for thrust outside the available range.

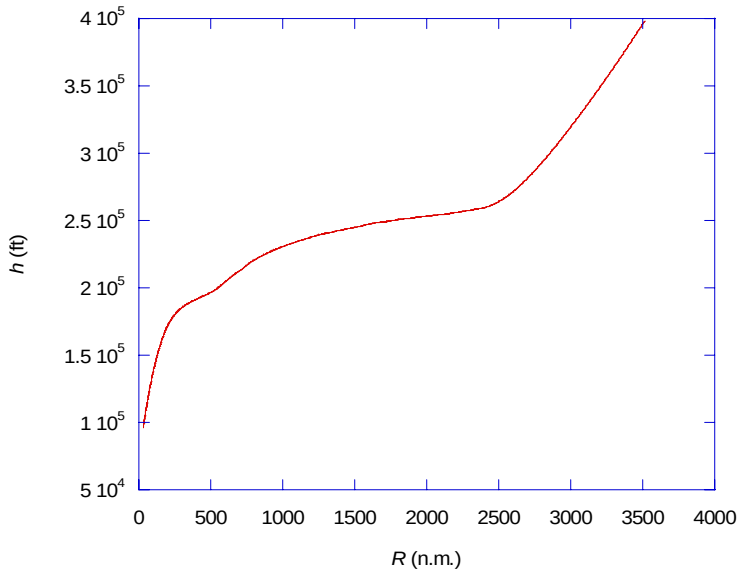

Figure 8-1: Entry altitude vs range-to-go on a typical entry trajectory

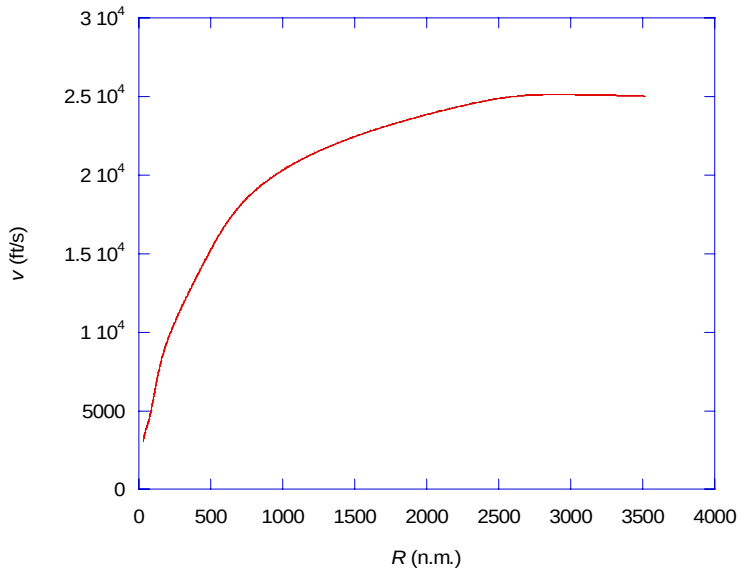

Figure 8-2: Entry speed vs range-to-go on a typical entry trajectory

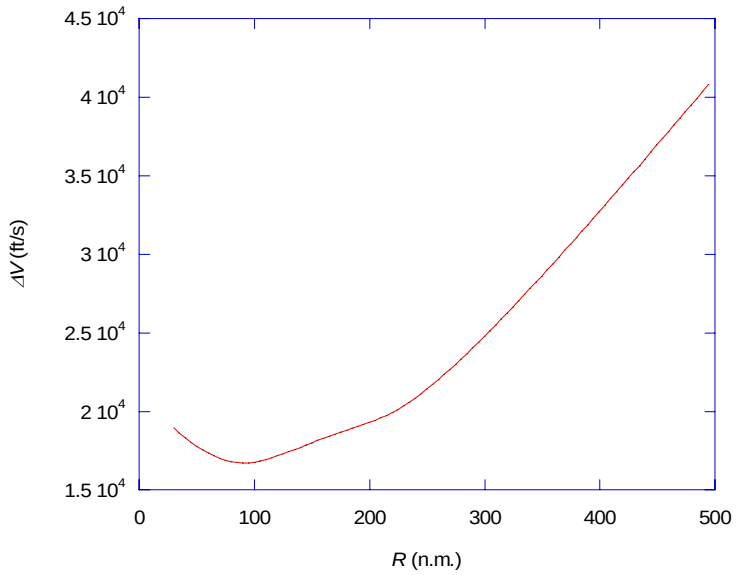

Figure 8-3: ΔV Required vs range-to-go for RTLS Trajectories

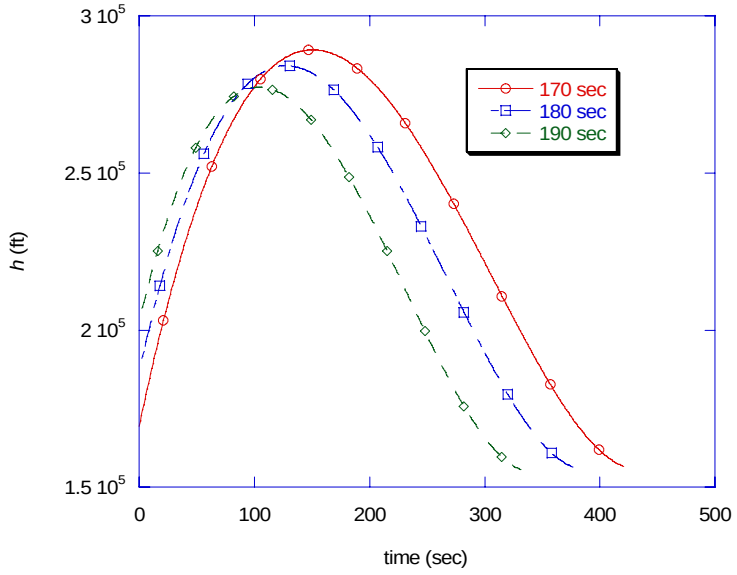

Figure 8-4: Altitude vs time on Open-loop RTLS Trajectories

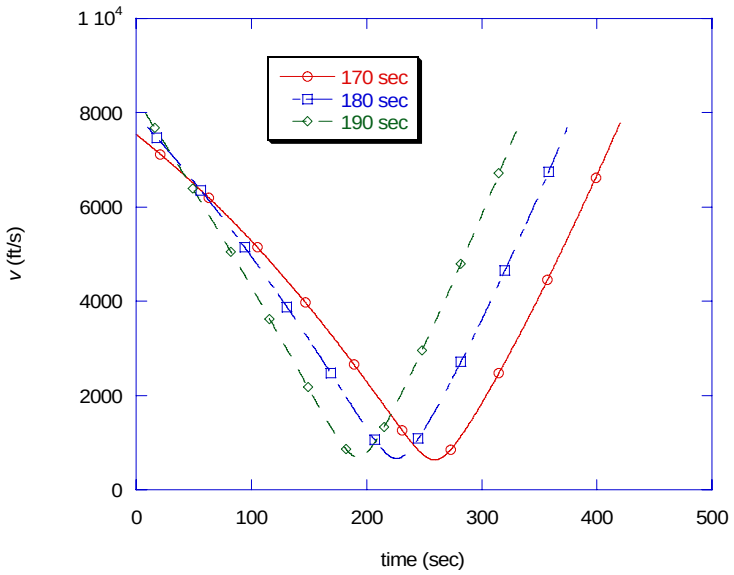

Figure 8-5: Speed vs time on Open-loop RTLS Trajectories

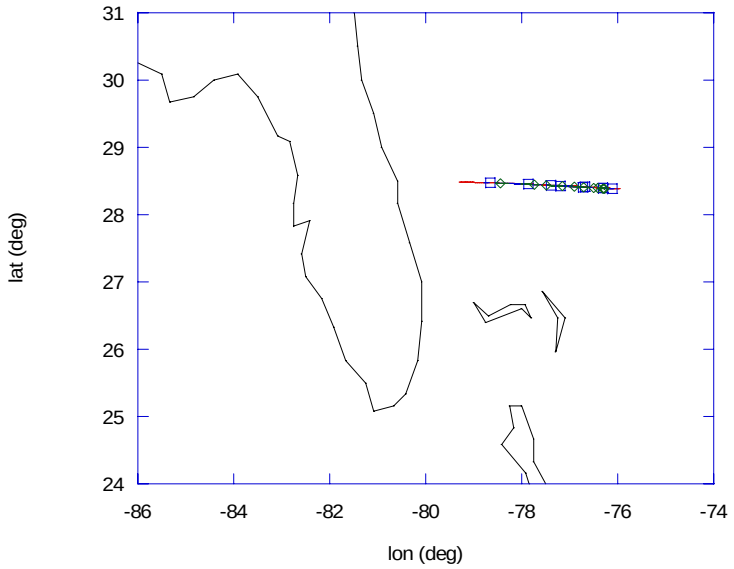

Figure 8-6: Ground Tracks on Open-loop RTLS Trajectories

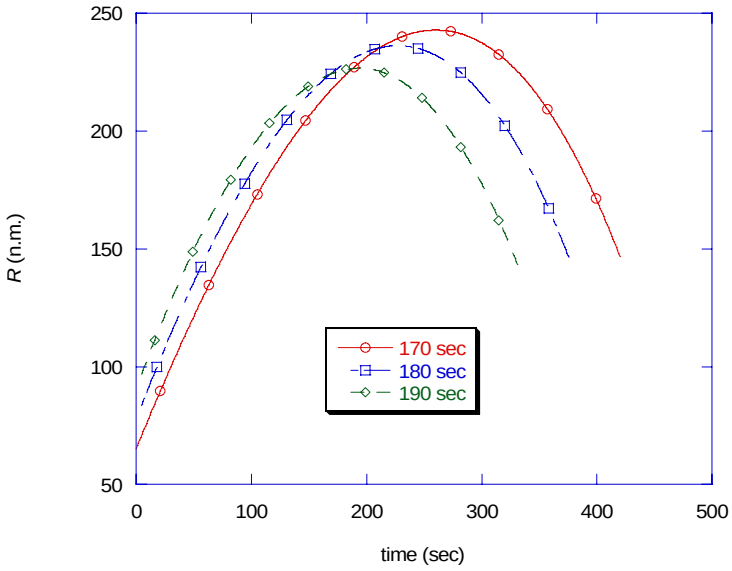

Figure 8-7: Down range vs time on Open-loop RTLS Trajectories

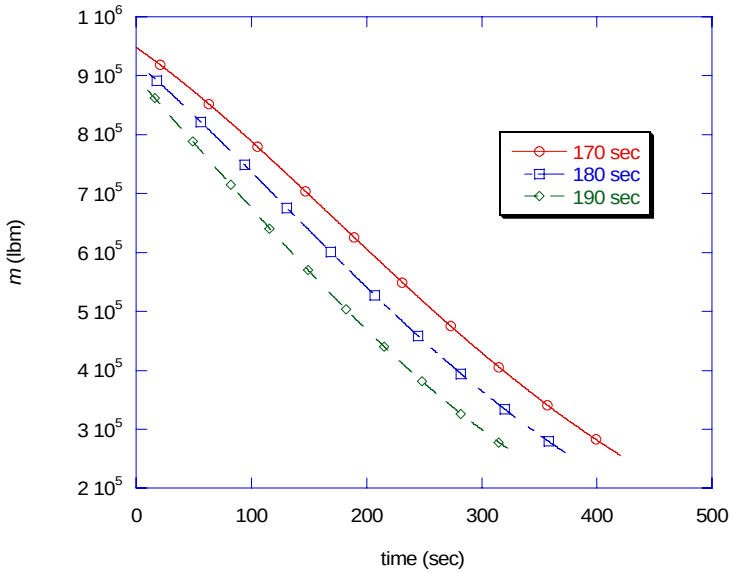

Figure 8-8: Mass vs time on Open-loop RTLS Trajectories

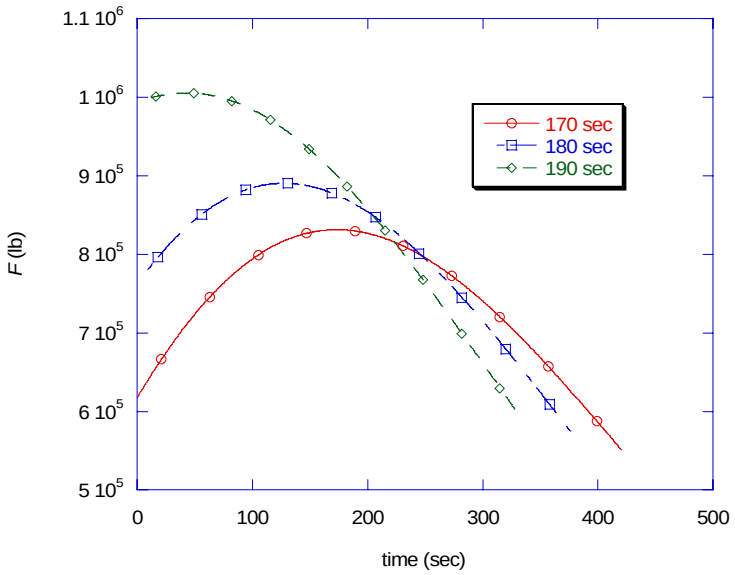

Figure 8-9: Thrust vs time on Open-loop RTLS Trajectories

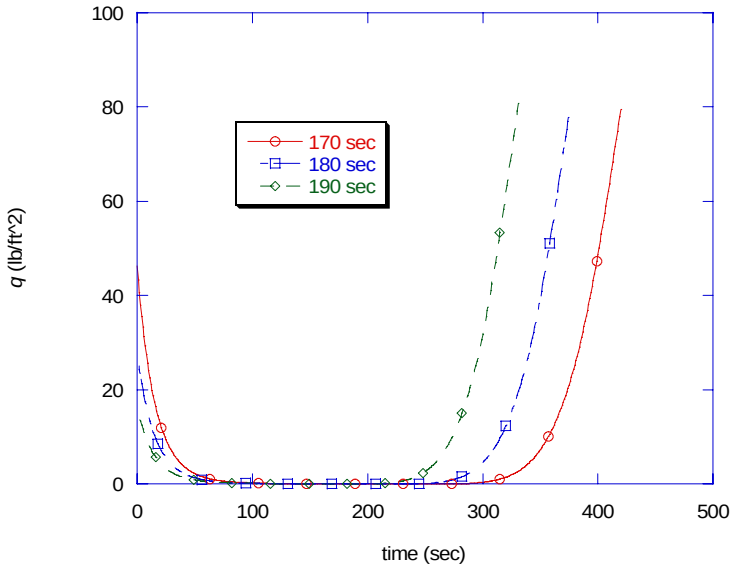

Figure 8-10: Dynamic pressure vs time on Open-loop RTLS Trajectories

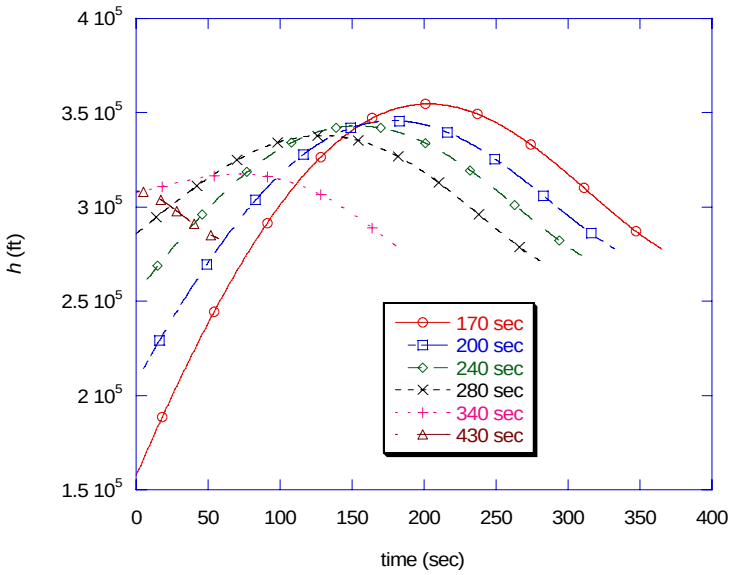

Figure 8-11: Altitude vs time on Open-loop ATDS Trajectories

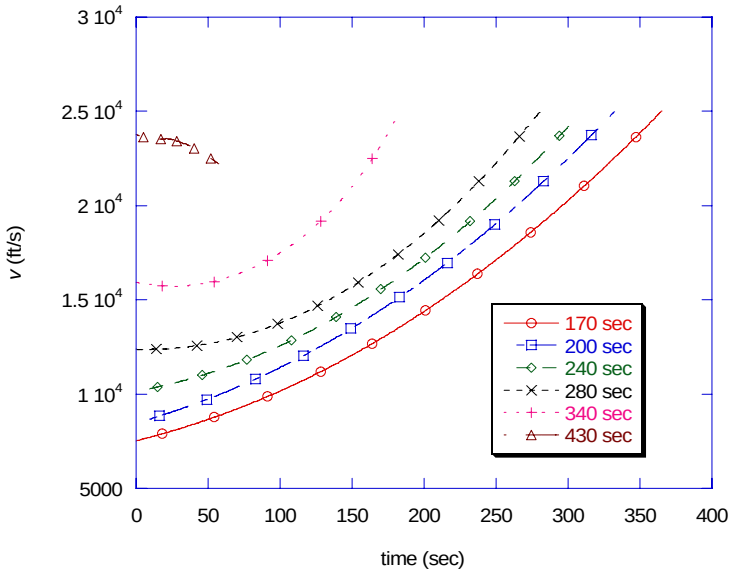

Figure 8-12: Speed vs time on Open-loop ATDS Trajectories

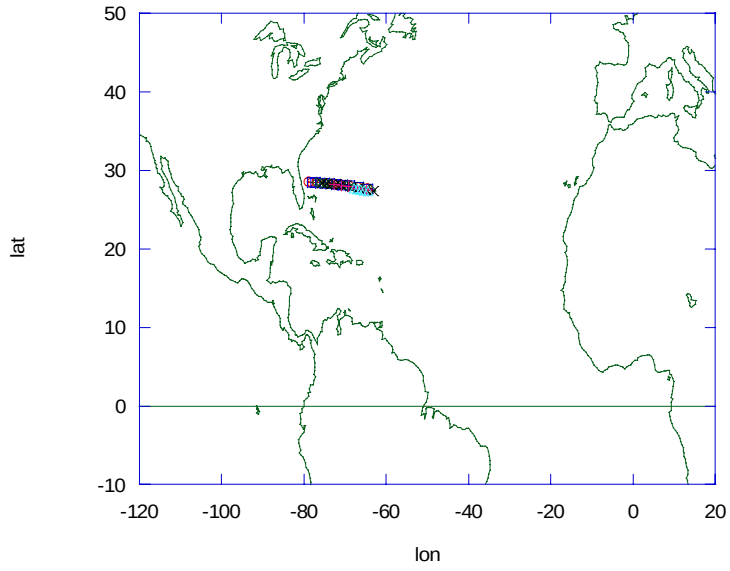

Figure 8-13: Ground Tracks on Open-loop ATDS Trajectories

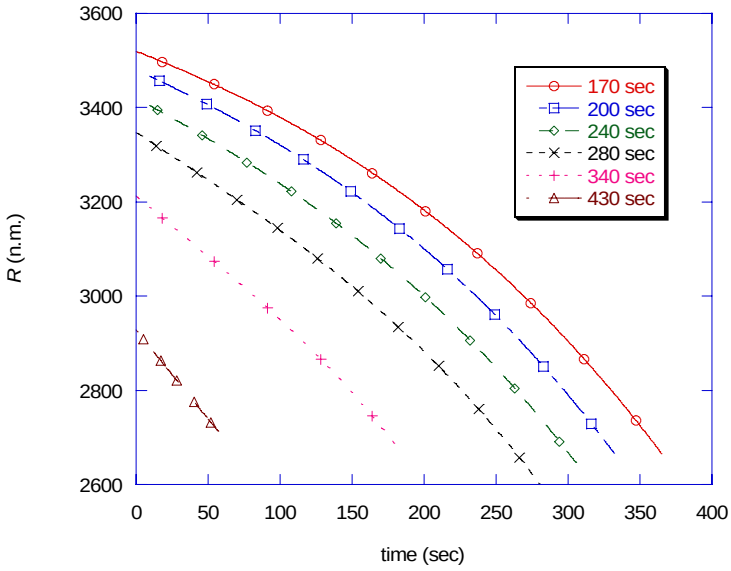

Figure 8-14: Down range vs time on Open-loop ATDS Trajectories

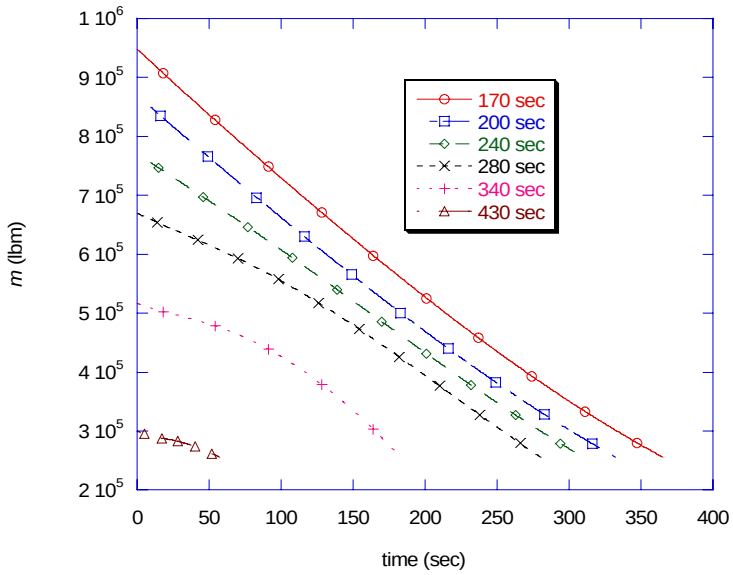

Figure 8-15: Mass vs time on Open-loop ATDS Trajectories

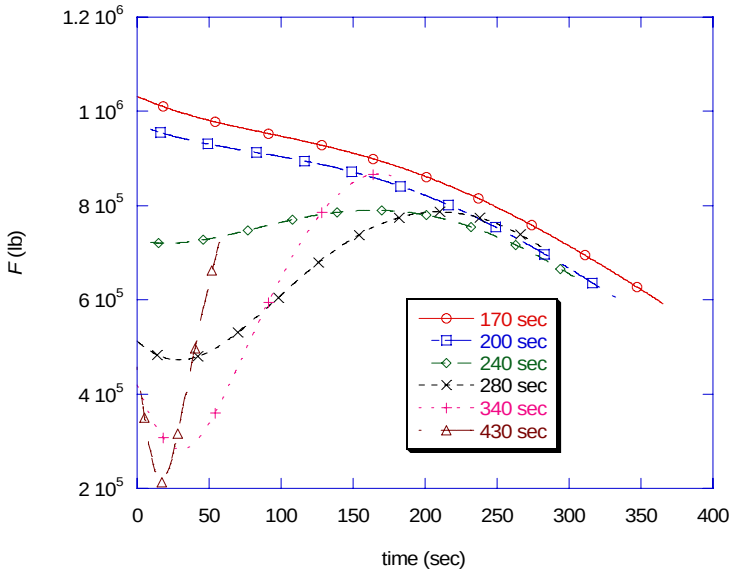

Figure 8-16: Thrust vs time on Open-loop ATDS Trajectories

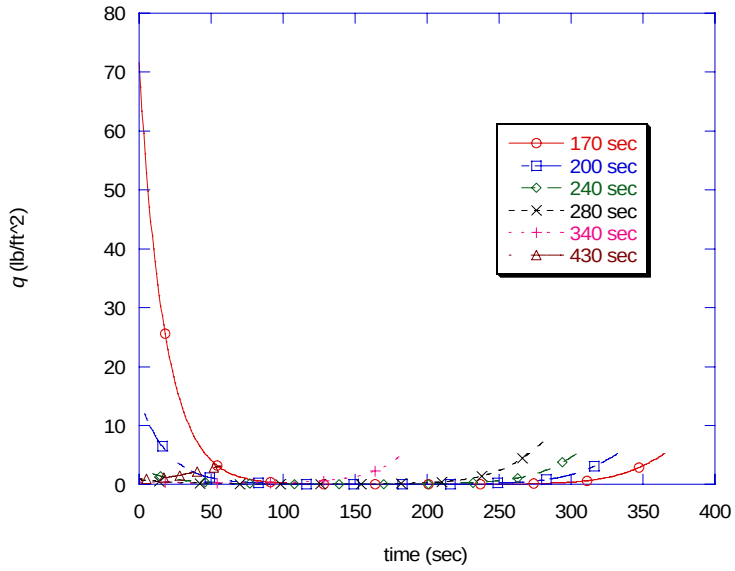

Figure 8-17: Dynamic pressure vs time on Open-loop ATDS Trajectories

CHAPTER 9

CLOSED-LOOP ABORT GUIDANCE

The previous chapter described how to generate abort trajectories that satisfy boundary conditions on the states and the fuel-depletion constraint while staying within vehicle thrust capabilities to the extent possible. This chapter addresses the problem of determining a viable closed-loop guidance protocol. Cyclical re-computation of the trajectories with the latest vehicle navigation data results in a candidate closed-loop guidance scheme for flying the generated trajectories. This is the approach used onboard the space shuttle [3]. As an alternative to re-computing the trajectories to effect closed-loop guidance, a closed-loop feedback technique is adopted here that results in stable tracking of the reference (i.e., most-recently generated) trajectory.

9.1 General Development

There are two possible methods for implementing closed-loop guidance. The first is to cyclically re-compute abort trajectories each guidance cycle based on the latest navigation state data and simply command the vehicle to fly the new "initial" acceleration vector. A potential problem with this approach is that the abort trajectory problem is very nonlinear, especially the RTLS trajectories. The latest navigation state estimates, unless the vehicle is performing very

near nominally, could cause relatively large perturbations in the newly computed abort trajectory (based on the new navigation data) resulting in guidance commands that are not very smooth. This in turn could cause larger guidance command variations than necessary.

Two considerations work to our favor here. First, unlike in the case of nominal ascent guidance, for example, fuel optimality is not as big a concern. There is no need to be concerned with cyclically re-optimizing the remaining trajectory - in the abort scenario, optimal payload performance is not a driving concern. Second, end-condition accuracy is not as critical as in the nominal ascent case. It is known from experience that entry guidance is very robust to initial entry condition errors (these are the powered guidance end-condition errors), therefore, the error tolerance (achieved state relative to target state) is relatively large. Thus, the choice is made to implement a guidance protocol wherein an abort trajectory is generated just once, at the time of vehicle health (or engine) failure, and subsequently track the "reference" trajectory for the rest of the powered flight. A new abort trajectory is generated if subsequently a new problem occurs on the vehicle, e.g., another engine goes out. A couple of side benefits to this guidance approach are reduced computational load and elimination of classical guidance sensitivity as time-to-go becomes small.

Define the position error, $\delta\mathbf{r}(t)$

$$\delta\mathbf{r} \equiv \mathbf{r}_{nav}(t) - \mathbf{r}_{ref}(t) \tag{9-1}$$

where $\mathbf{r}_{ref}$ is taken from the cubic polynomial. Now construct a standard second-order system with the position error the variable to be controlled:

$$\delta\ddot{\mathbf{r}}(t) + 2\zeta\omega_n\delta\dot{\mathbf{r}}(t) + \omega_n^2\delta\mathbf{r}(t) = \mathbf{0} \tag{9-2}$$

The first and second time derivatives of position error are given by:

$$\delta\dot{\mathbf{r}}(t) = \mathbf{v}_r(t) - \mathbf{v}_{ref}(t) \equiv \delta\mathbf{v}_r \tag{9-3}$$

$$\delta\ddot{\mathbf{r}}(t) = \mathbf{a}(t) - \mathbf{a}_{ref}(t) \tag{9-4}$$

where $\mathbf{a}_{\text{ref}}$ is evaluated from the linear polynomial. Substituting the previous into the second-order system equation and solving for the relative acceleration, **a**, gives:

$$\mathbf{a}(t) = \mathbf{a}_{ref}(t) - 2\zeta\omega_n\delta\mathbf{v}_r - \omega_n^2\delta\mathbf{r} \tag{9-5}$$

Recall that relative acceleration consists of propulsive acceleration, gravity acceleration, and Coriolis acceleration – solving for "commanded" propulsive acceleration, $\mathbf{a}_{p\text{-}cmd}$, yields:

$$\mathbf{a}_{p-cmd} = \mathbf{a}_{ref}(t) - 2\zeta\omega_n\delta\mathbf{v}_r - \omega_n^2\delta\mathbf{r} + \frac{\mu\mathbf{r}_{nav}}{|\mathbf{r}_{nav}|^3} + 2(\boldsymbol{\omega}\times\mathbf{v}_{r-nav}) \tag{9-6}$$

One way of implementing the commanded acceleration magnitude is to use estimated vehicle mass and known maximum thrust to calculate a throttle command. Alternatively, a feedback loop can be designed to use accelerometer feedback and adjust the throttle to track the commanded acceleration magnitude. There may be saturation periods where the commanded acceleration is greater than the maximum attainable thrust or less than the minimum attainable thrust. This is

usually not harmful as long as the period of saturation does not last too long compared to the total flight time.

The next section describes a technique for improving reference trajectory tracking, particularly when the vehicle is performing in an off-nominal manner or when the throttle command is saturated.

9.2 Choice of Independent Variable

The reference trajectory tracking guidance scheme requires the ability to calculate reference trajectory parameters as a function of some independent variable. The first variable that comes to mind is time. However, time contains no information about the vehicle state. Unless guidance command implementation and navigation information is perfect, the vehicle will deviate from the nominal trajectory, and tracking will tend to get progressively worse and result in significant errors at burnout. The preferred type of independent variable is one which contains some information about how the vehicle is performing - in effect this provides some measure of state feedback. An example is space shuttle first stage ascent guidance which uses Earth-relative speed as the independent variable in the pre-stored attitude tables. The algorithm designers could have used, say, time from liftoff, but the trajectory dispersions at SRB separation would be unacceptably large.

Our choice of independent variables in the abort guidance application is limited because states such as speed, altitude, energy are generally non-monotonic on abort trajectories. Propellant consumed since initiation of the abort trajectory is monotonic. Another good choice would be ΔV imparted since abort initiation. The latter is chosen because it is easy to estimate and sum from navigation measurements. Hypothetically, the reference trajectory as a function of

ΔV imparted could be computed and stored. A simpler method is outlined here. Introduce the "time-to-go" which is related to the duration of powered flight, T, and the time from abort trajectory initiation, t:

$$t_{go} = T - t \tag{9-7}$$

On the first guidance cycle, the abort trajectory is generated and the time-to-go is initialized to the value of T. On subsequent guidance cycles, the time-to-go is decremented by the guidance cycle time, Δt_{guid}, scaled by the ratio of sensed ΔV to nominal (expected) ΔV:

$$t_{go} := t_{go} - \Delta t_{guid} \frac{\left| \mathbf{v}_{r-nav} - \mathbf{v}_{r-nav-prev} \right|}{\left| \mathbf{v}_{ref}\left(T - t_{g0}\right) - \mathbf{v}_{ref}\left(T - t_{go} - \Delta t_{guid}\right) \right|} \tag{9-8}$$

where $\mathbf{v}_{r\text{-}nav\text{-}prev}$ is the vehicle's Earth-relative velocity vector (stored) from the previous guidance cycle.

9.3 Guided Trajectory Results

The trajectory generation protocol described in Chapter 8 was used to generate feasible RTLS and ATDS trajectories. The abort times simulated here are the same as those of Chapter 8. Optimization objective (8-21), i.e., minimize the integral of the square of thrust minus desired thrust, was used. The reference trajectory tracking guidance and steering technique described above was used. Damping ratio was set to 0.7 and undamped natural frequency set to 1.25 / T.

Minimum commanded thrust was limited to 500,000 lb in the simulation with an upper limit of 1,000,000 lb. Guidance was executed at a rate of 10 Hz. A total maneuver rate constraint of 15 dps was imposed. The initial angles of attack and sideslip were taken to be zero deg in all cases to simulate the initial attitude not being equal to that of the initial open-loop attitude.

Figures 9-1 through 9-12 show plots of the RTLS guided trajectories. Figures 9-1 and 9-2 show the altitude and Earth-relative velocity histories, resp. As in the open loop trajectories of Chapter 8, the final altitudes and speeds are similar. The signature drop in speed after the turn around is evident as the vehicle performs a retrograde burn to arrest its down range motion. The final down range, shown in Figure 9-3, is between 150 and 160 n.m. Figure 9-4 shows that the vehicle burns all of its fuel on all the trajectories. Figure 9-5 shows the actual thrust vs. time on the RTLS trajectories. The (late) 190 s RTLS abort trajectory has an extended thrust saturation interval of about 120 s. Figures 9-6 through 9-8 show the dynamic pressure, load indicators $q\alpha$, and $q\beta$, angle of attack and sideslip curves. The $q\alpha$ values approach -3,500 lb per square foot deg on the earliest RTLS abort. This could be mitigated if need be by reducing the pitch maneuver rate during the turnaround. The angles of sideslip are small throughout due to the planar nature of RTLS trajectories. Figure 9-9 shows the time-to-go vs. time, computed as discussed in this chapter. The linear nature of time-to-go indicates that the effects of thrust saturation are not too significant or long-lived to be deleterious.

Heading error is shown in Figure 9-10. Heading error is the velocity heading angle minus the bearing angle to the targeted landing site. A value of zero indicates that the vehicle is heading directly for the landing site. Figures 9-11 and 9-12 show the altitude and speed error. The values at any instant represent the errors, with respect to the entry profile, that the vehicle would have if the powered maneuver were terminated at that instant. The figures indicate that the 180 s abort case ends up high in altitude by about 15,000 ft but low in speed by about 300 fps. Although

such errors are undesirable, the low speed compensates (vis a vis desired energy vs. range relationship) for the high altitude, reducing the effect of each individual error. The attitude rate limits during the turn around maneuver are the main cause of the errors in that rate limits have a direct effect on trajectory control.

Figures 9-13 through 9-25 show plots of the ATDS guided trajectories. Figures 9-13 and 9-14 show the altitude and Earth-relative velocity histories, resp. Again, the final altitudes and speeds are similar to one another. The final down range, shown in Figure 9-15, is between 2,600 and 2,800 n.m. in all cases. Figure 9-16 shows that the vehicle burns all of its fuel on all the trajectories. Figure 9-17 shows the actual thrust vs. time on the ATDS trajectories. The late abort trajectories involve significant periods of thrust saturation. This is consistent with the respective open-loop trajectories. Figures 9-18 through 9-21 show the dynamic pressure, load indicators $q\alpha$, and $q\beta$, angle of attack and sideslip curves. The loads look very benign with the worst-case $q\alpha$ of 1,500 psf-deg occurring for the earliest abort due to the relatively high dynamic pressure present at the time of abort.

The angles of sideslip approach 90 deg for the late abort cases because late in nominal ascent the vehicle has high speed, so any off ground track targeting at all requires significant out-of-plane steering. Moreover, the required out-of-plane trajectory change is no longer dominated by the in-plane trajectory change. It makes sense in such cases to relax the heading error constraint and let entry guidance null out small heading errors on the order of a few degrees. This could be done with a modification to the trajectory generation procedure, i.e., target vectors construction logic. Figure 9-22 shows the time-to-go vs. time, computed as discussed in this chapter. The time-to-go is not very linear for late abort times, due to large thrust saturation periods. Heading error is shown in Figure 9-23. In all cases, the final heading error is reduced to less than 2 deg, which is well within the capability of entry guidance. In cases where there are not excessive

amounts of thrust saturation, i.e., on the early aborts, the final heading errors are a small fraction of a degree. Figures 9-24 and 9-25 show the altitude and speed errors. The worst-case altitude errors are about 5,000 ft with worst-case speed errors of about 500 fps, well within the capability of entry guidance, particularly considering the long range-to-go at powered maneuver termination. The worst-case speed error of 500 fps occurs for the latest abort (abort time of 430 s) due to a combination of short burn time and saturated thrust throughout the entire maneuver.

The guidance protocol developed and demonstrated here provides satisfactory results for those cases tested. The simulation used to generate the guided results did not include aerodynamic forces. Because dynamic pressure is so small during the powered abort maneuver, the effects of aerodynamics are expected to be small. The main effect would be a reduction in energy at the cutoff point due to drag. This could be compensated for in the trajectory generation process by reducing, or biasing, the amount of remaining ΔV by a small amount. This could be used to provide some "energy" margin to account for unmodeled effects in the trajectory generation process. Another area for study is in the guidance law. Stable tracking is only guaranteed in the absence of thrust saturation. Modifications to the guidance law could potentially provide better trajectory tracking in the presence of thrust saturation.

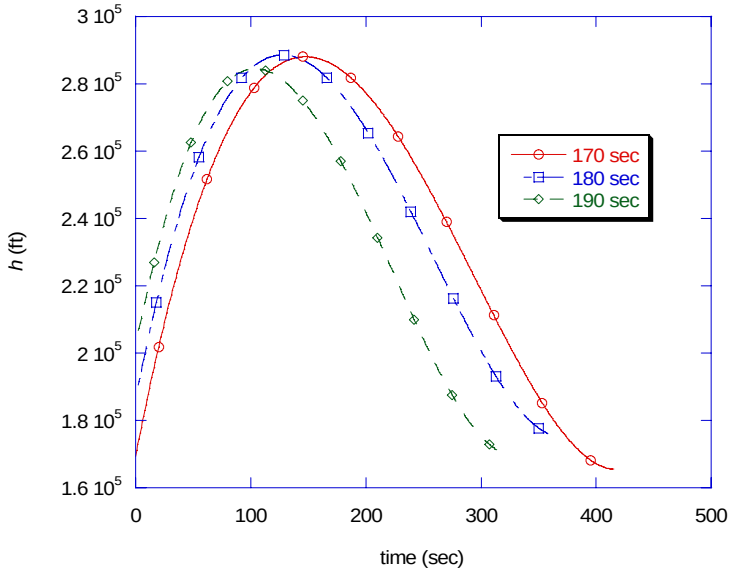

Figure 9-1: Altitude vs time on Guided RTLS Trajectories

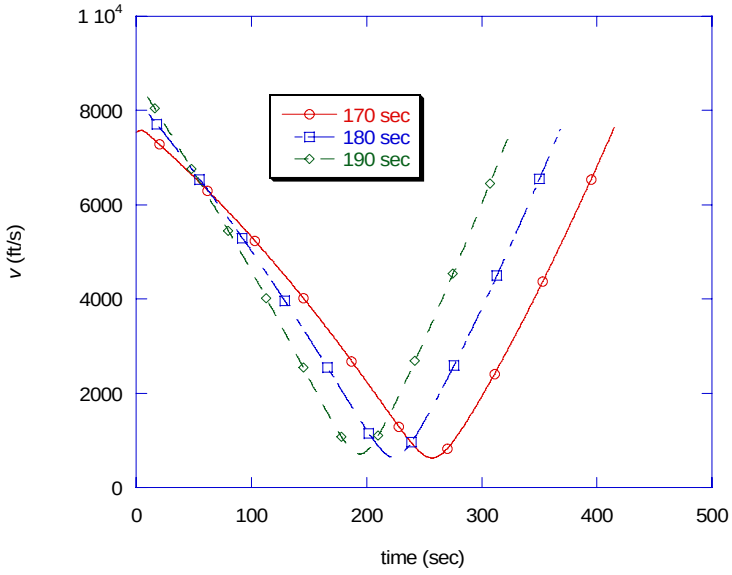

Figure 9-2: Speed vs time on Guided RTLS Trajectories

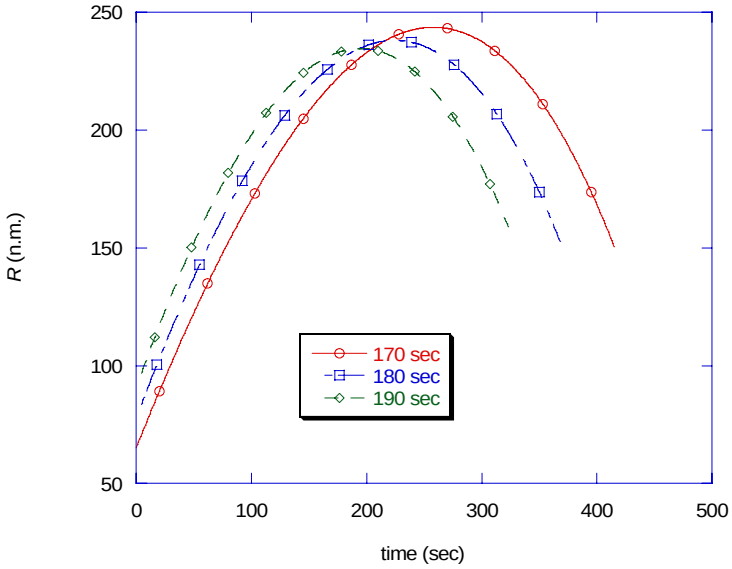

Figure 9-3: Down range vs time on Guided RTLS Trajectories

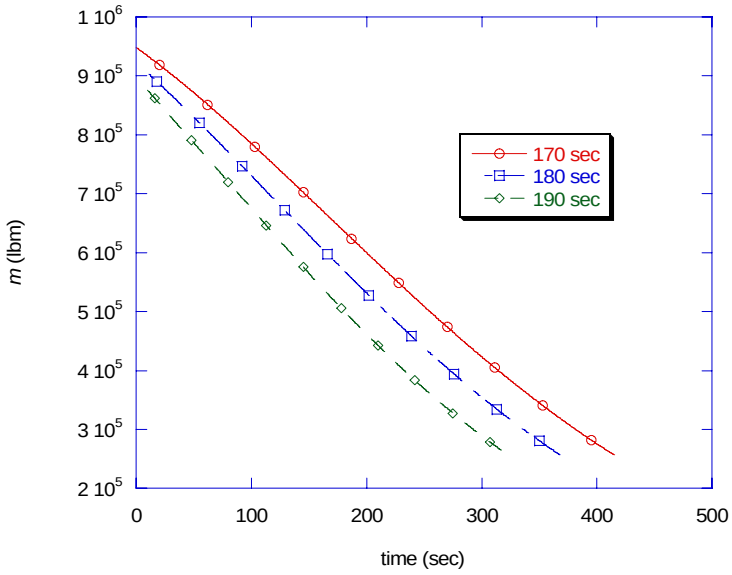

Figure 9-4: Mass vs time on RTLS Guided Trajectories

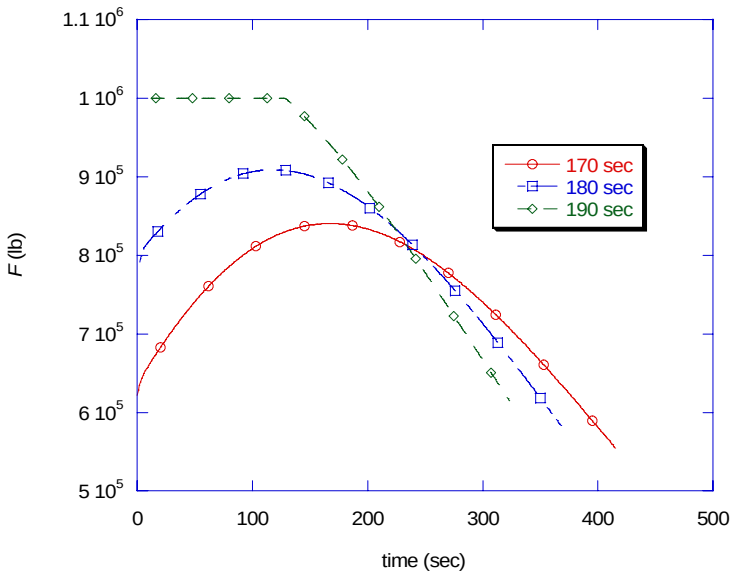

Figure 9-5: Thrust vs time on Guided RTLS Trajectories

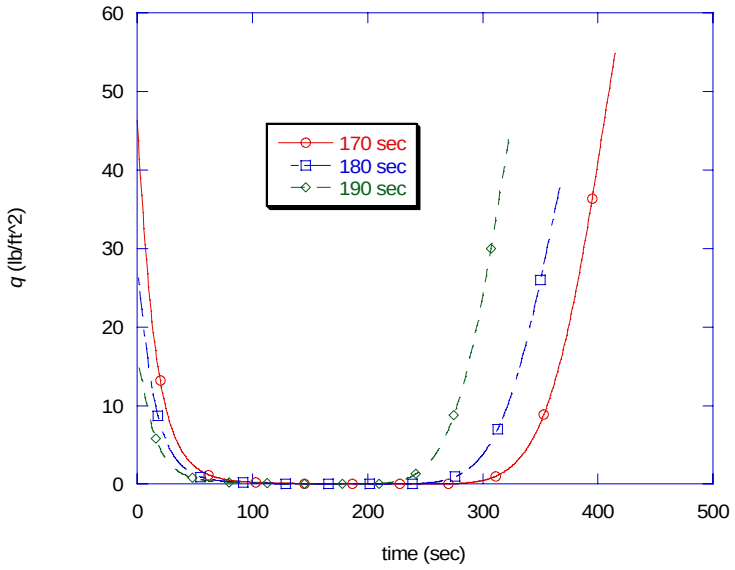

Figure 9-6: Dynamic pressure vs time on Guided RTLS Trajectories

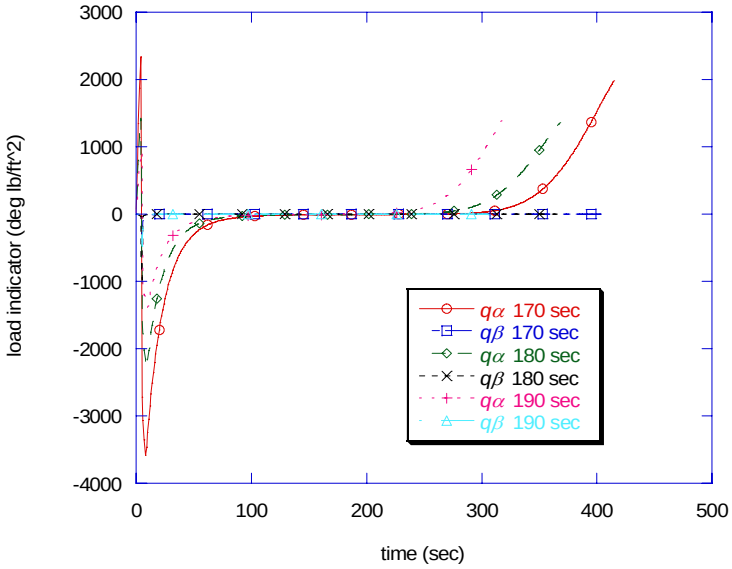

Figure 9-7: $q\alpha$, $q\beta$ vs time on Guided RTLS Trajectories

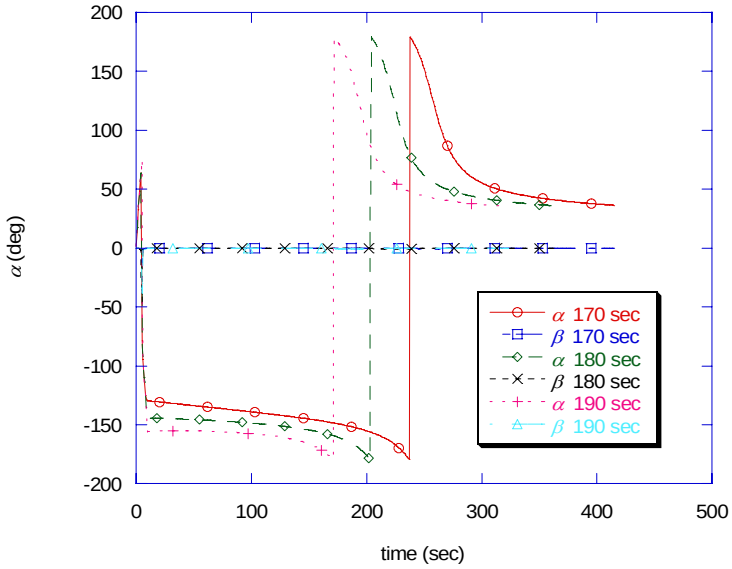

Figure 9-8: α, β vs time on Guided RTLS Trajectories

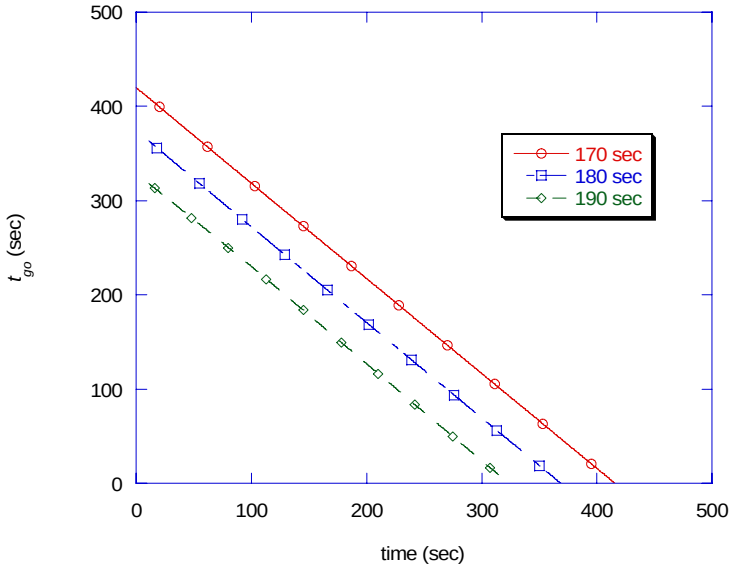

Figure 9-9: Time-to-go vs time on Guided RTLS Trajectories

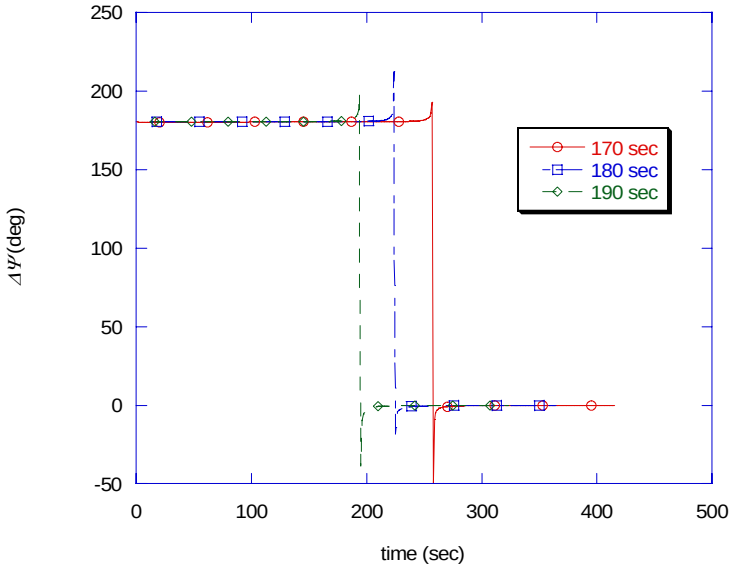

Figure 9-10: Heading error vs time on Guided RTLS Trajectories

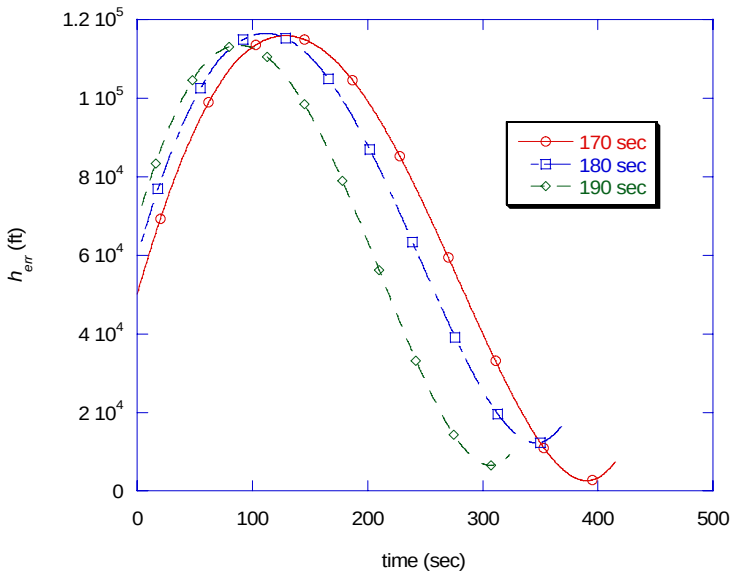

Figure 9-11: Altitude error vs time on Guided RTLS Trajectories

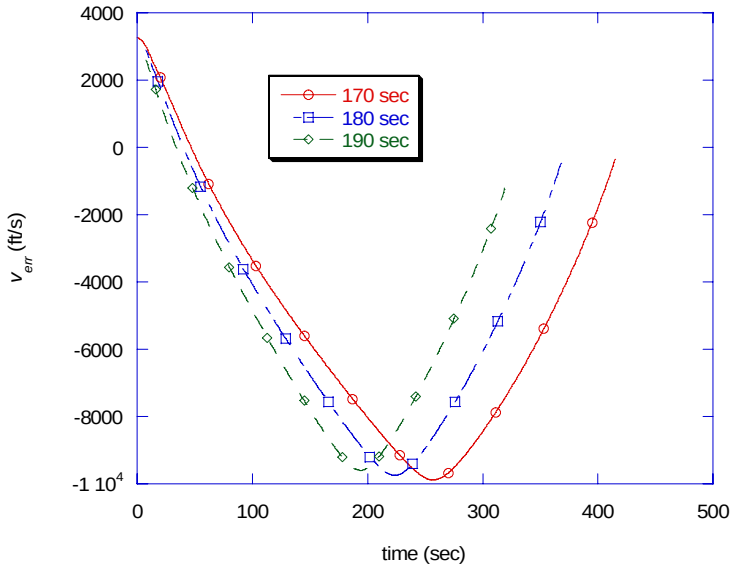

Figure 9-12: Speed error vs time on Guided RTLS Trajectories

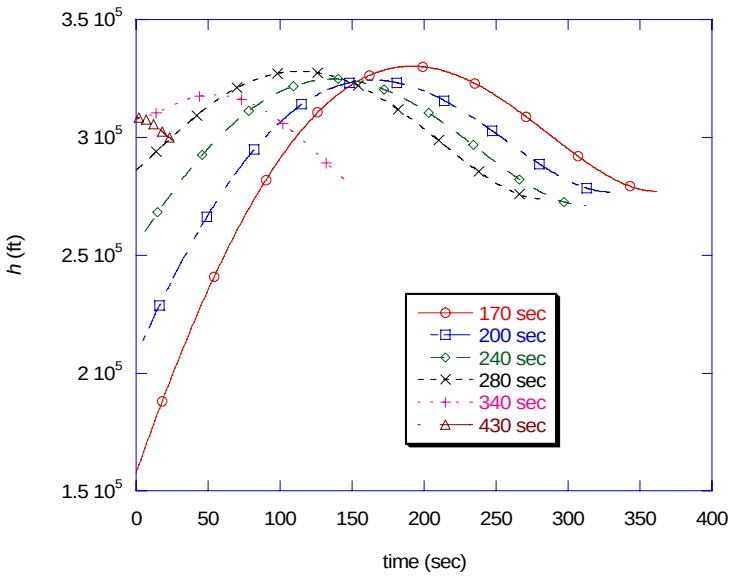

Figure 9-13: Altitude vs time on Guided ATDS Trajectories

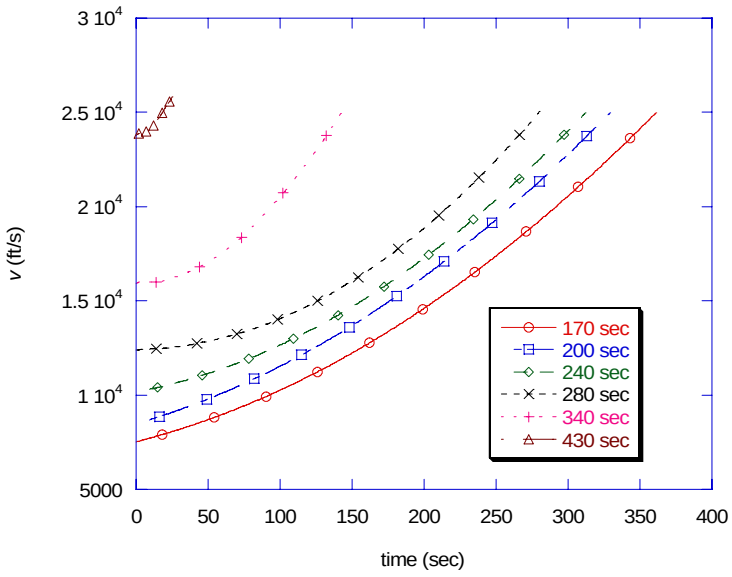

Figure 9-14: Speed vs time on Guided ATDS Trajectories

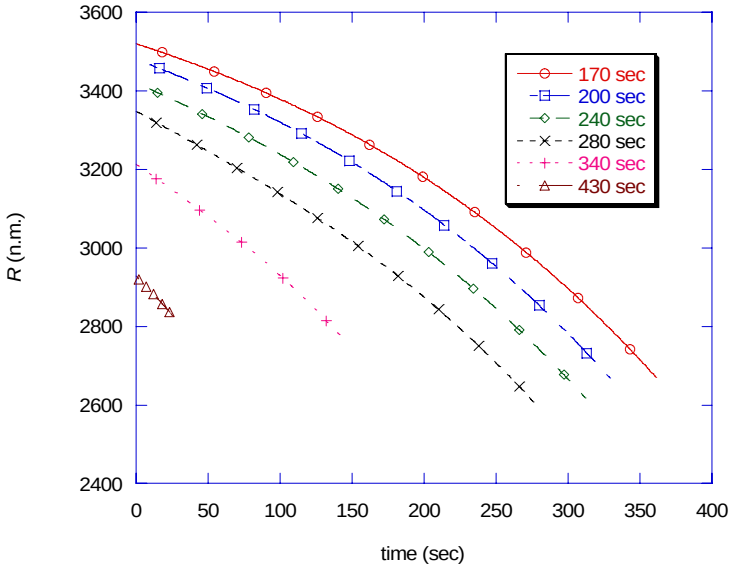

Figure 9-15: Down range vs time on Guided ATDS Trajectories

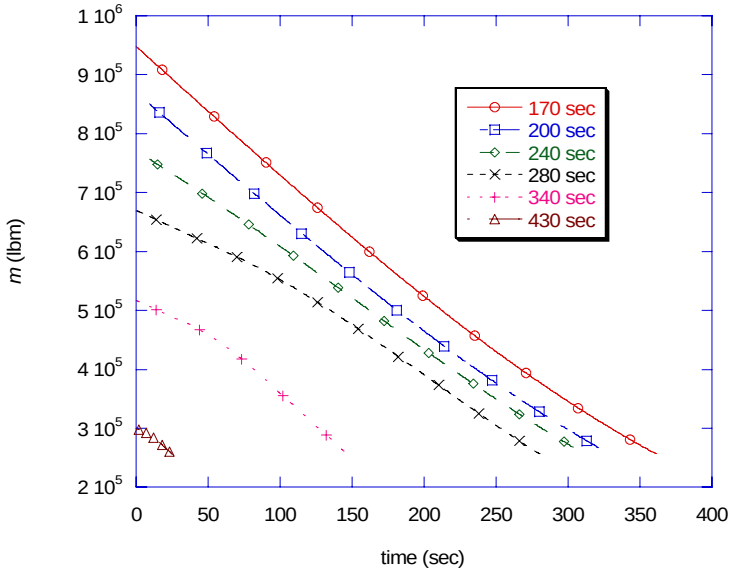

Figure 9-16: Mass vs time on Guided ATDS Trajectories

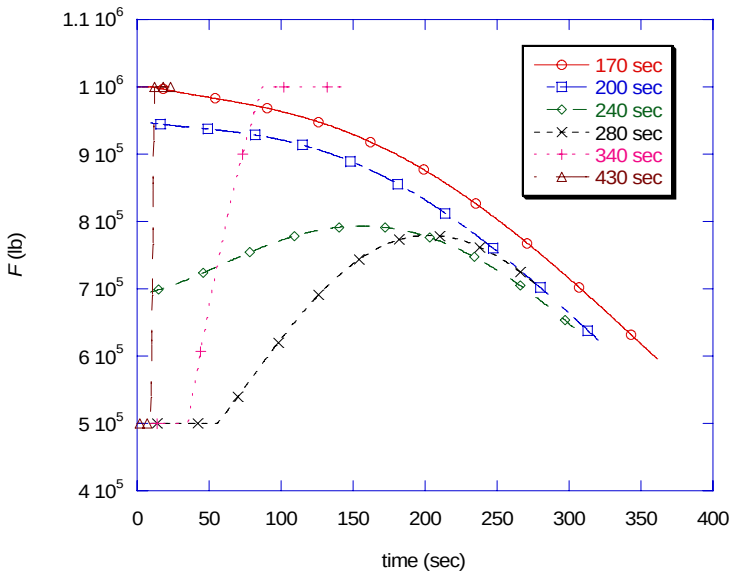

Figure 9-17: Thrust vs time on Guided ATDS Trajectories

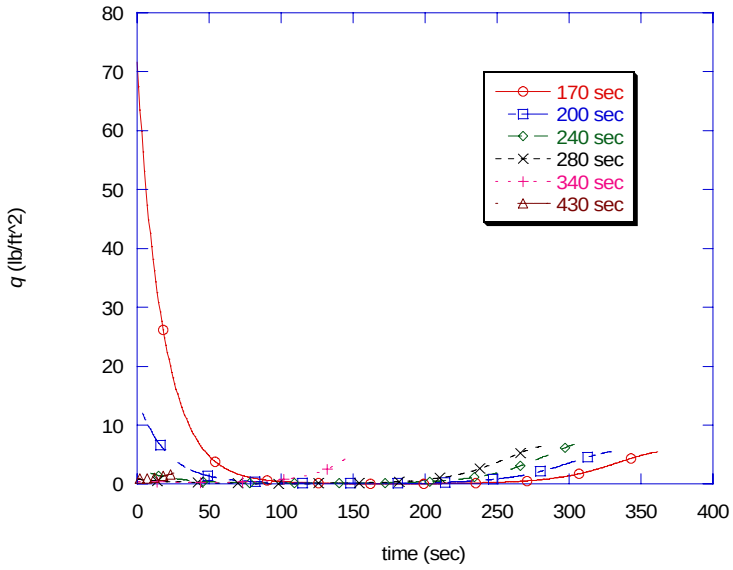

Figure 9-18: Dynamic pressure vs time on Guided ATDS Trajectories

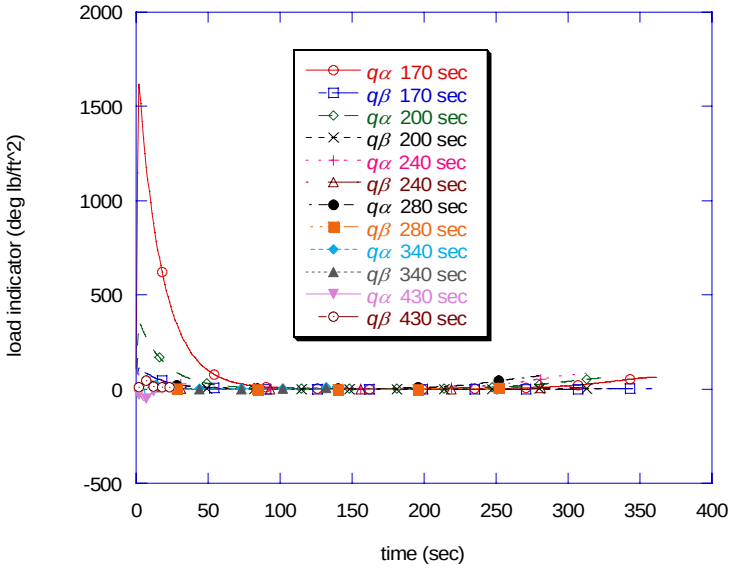

Figure 9-19: $q\alpha$, $q\beta$ vs time on Guided ATDS Trajectories

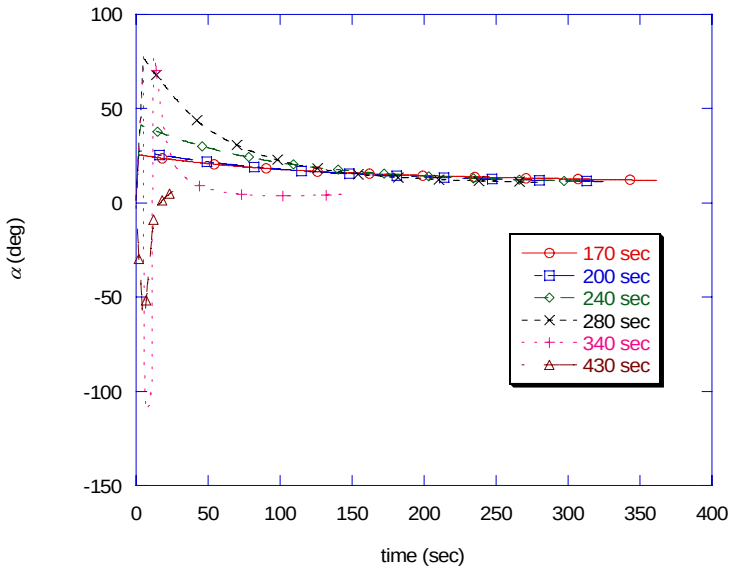

Figure 9-20: α vs time on Guided ATDS Trajectories

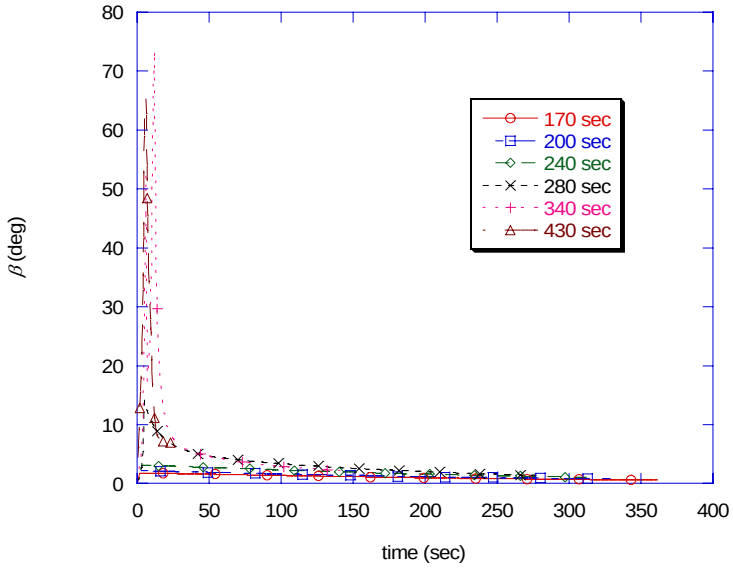

Figure 9-21: β vs time on Guided ATDS Trajectories

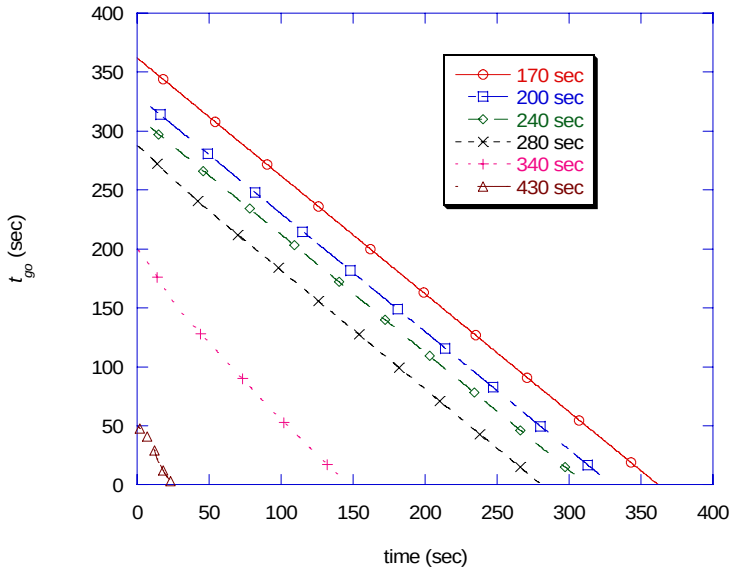

Figure 9-22: Time-to-go vs time on Guided ATDS Trajectories

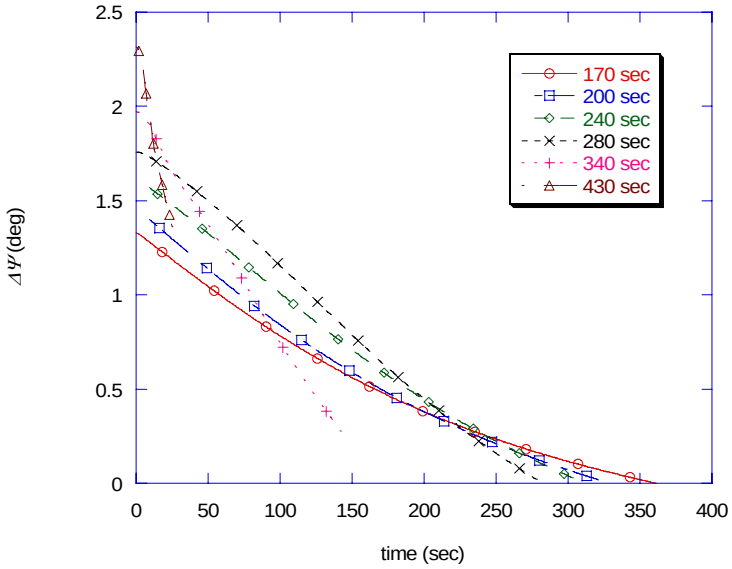

Figure 9-23: Heading error vs time on Guided ATDS Trajectories

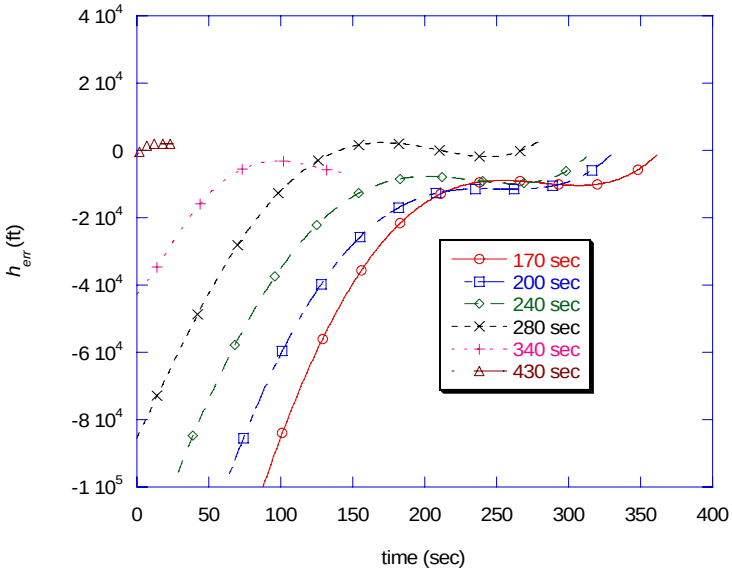

Figure 9-24: Altitude Error vs time on Guided ATDS Trajectories

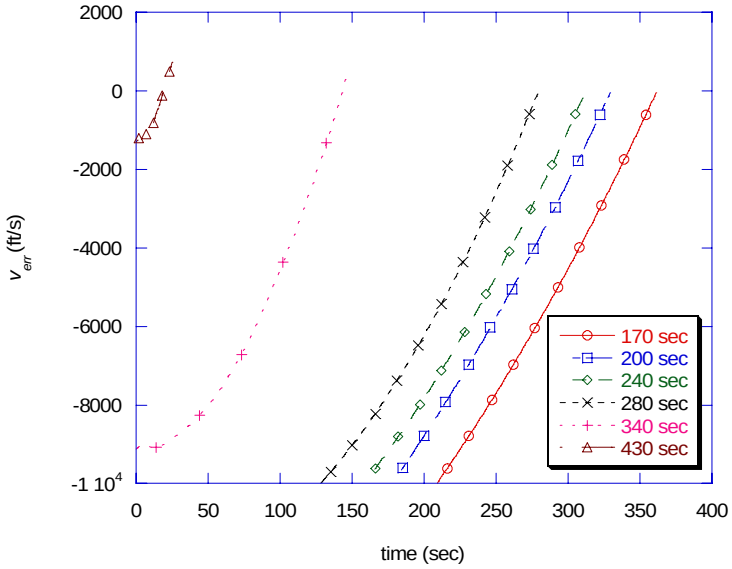

Figure 9-25: Speed error vs time on Guided ATDS Trajectories

CHAPTER 10

CONCLUSIONS AND FUTURE RESEARCH

This thesis treated various aspects of the atmospheric ascent trajectory generation and guidance problem. The intent was to contribute to the state of the art of ascent guidance so that future launch vehicles will be more adaptive and less dependent on pre-mission analyses for nominal and abort trajectories. This should help to reduce operations costs and increase flight safety.

The main contributions are summarized below.

(1) The nominal ascent trajectory generation problem, including the high dynamic pressure flight phase, was treated. The formulation addressed the common wings-level flight protocol. A modified multiple shooting method was used to solve the two-point boundary value problem resulting from application of the Euler-Lagrange conditions. This numerical method has the salient feature of guaranteed (quadratic) convergence when the guessed parameters are all within some neighborhood of the solution. This is in contrast to, say, fixed-point iteration based algorithms, which have no guarantee of convergence even when very close to the solution. The favorable convergence properties of multiple shooting suggest the use of a cyclical re-optimization protocol to effect closed-loop guidance. This was demonstrated via presentation of guided trajectory results.

(2) The multi-arc trajectory optimization formulation was presented with strengthened necessary conditions of optimality relative to those found in the literature. Mission scenarios under which the strengthened conditions are equivalent to the weak conditions were presented. A fundamental discussion of burn-coast-burn trajectories was presented with the intent of providing physical understanding of why burn-coast-burn trajectories are more optimal than single-burn trajectories and also to be able to anticipate, according to the orbit transfer objectives, when the burn-coast-burn protocol would provide relative benefit.

(3) A modified multiple shooting method was applied to the burn-coast-burn ascent trajectory generation and guidance problem. Guided trajectories were presented. Various sets of terminal state constraints were formulated and tested in guided trajectory simulations. Strengths and weaknesses of various constraints were discussed.

(4) Analytical treatment of transversality conditions was presented, with the intent of providing insight into the nature of these necessary conditions which tend to add to the complexity of optimal control based methods. A numerical method for handling the transversality conditions was presented which relieves the guidance practitioner of this analytical burden and enables a menu-based protocol for specifying terminal state constraints in the guidance code. With this approach, commonly used terminal state constraint expressions can be coded individually and then combined arbitrarily (as long as the specified constraints combine to form linearly independent sets) to define a given set of mission objectives.

(5) The “return-to-launch-site” and “abort-to-down-range-site” trajectory generation problems were treated. The methods have an extra degree of freedom in that throttling is inherently taken into account and, hence, perhaps possess enhanced capability relative to the standard fixed thrust trajectory generation methods. When a solution exists, converge

is guaranteed. When a solution does not exist, the method reliably indicates that this is the case. Traditional methods can be ambiguous in that when they do not converge, there is uncertainty as to whether there is indeed no solution or the numerical method simply couldn't find a solution.

(6) A closed-loop abort guidance scheme was developed and demonstrated via guided trajectory results. The scheme is a reference trajectory tracking method which has significantly less computational burden than a cyclical re-optimization scheme. This choice of guidance scheme is driven by the fact that entry guidance schemes are very error tolerant and by the fact that optimization is not a primary concern for aborts (as it is for, say, nominal ascents.) Moreover, the classical guidance sensitivity as time-to-go becomes small is eliminated and guidance command sensitivity to various dispersions is reduced.

(7) The capability to model thrust components along the *z*-body axis and take this into account in the atmospheric trajectory generation process was developed. Although only a minor modification to the logic and formulation is required, this is an essential part of atmospheric ascent guidance technology because this has a significant effect on the trajectory. It was observed that the net engine gimbal angle can be modeled as a linear, or perhaps quadratic, depending on the vehicle mass properties, function of vehicle mass.

The overall launch vehicle trajectory generation and guidance problem is very involved. For obvious reasons, many of the complexities of the problem are still resolved via ground-based computers and pre-mission analyses and procedures. The research presented in this thesis builds on the solid research of the Apollo and space shuttle era guidance practitioners as well as recent researchers in the field. It is the hope of the author that the research reported

here and that of current researchers will, over time, lead to more capable and adaptive launch vehicles. Below are some recommendations for further research in this area.

(1) Investigate methods for handling dynamic pressure constraints within the trajectory generation process. Often, this is not an active constraint but it would be useful to be able to explicitly constrain maximum dynamic pressure when needed.

(2) Investigate the efficacy of a protocol involving one-time atmospheric trajectory optimization and a reference trajectory tracking scheme during the high dynamic pressure region. The trajectory generation would still be done on the vehicle but guidance would not re-optimize unless a large dispersion occurred or until after the high dynamic pressure region has passed. This modus operandi would have the advantage that the guidance would respond more smoothly than re-optimizing guidance which may tend to over respond to perturbations such as poorly modeled winds. Theoretically, better performance is obtained by re-optimizing, but this may not be the case in practice, or, perhaps, inconsequential from a practical standpoint.

(3) This technology needs to be exercised extensively in high-fidelity six degree of freedom trajectory simulators to verify and validate the logic and formulations. This should include comparison with space shuttle trajectory generation and guidance methods.

(4) It would be interesting to quantify the benefits of the abort trajectory generation relative to traditional fixed-thrust formulations. It would be interesting, for example, to see how much of a given vehicle capability envelope the two different methods can cover.

(5) No (global) convergence proofs exist at this time for the multiple shooting based methods developed in this thesis. The next best thing to a proof is to exhaustively exercise the algorithms and show that they converge for all anticipated sets of inputs. Although the algorithms have been exercised fairly extensively and have been successful, there is still a lot

more testing, and convincing, to do before such algorithms become serious candidates for onboard implementation.

(6) The abort formulation does not take into account aerodynamic forces and the guided trajectory results do not include aerodynamic forces. The inherent assumption used in the abort formulation is that a steering approach similar to that of Chapters 2 and 3, or some other approach, is used to steer the vehicle out of the high dynamic pressure region followed by the abort formulation of Chapters 8 and 9 to perform the exo-atmospheric portion of the abort maneuver. The presumption is that the dynamic pressure during the exo-atmospheric phase would be small enough so that the aerodynamic forces act as small perturbations to that portion of the trajectory. In principle, one could use the methods of Chapters 2 and 3 to compute the entire abort maneuver while taking into account aerodynamics, but it remains to be seen whether such methods can reliably and accurately solve (converge) for the nonlinear, highly-constrained maneuvers associated with the return-to-launch-site abort. The abort algorithms and methodology presented here need to be matured via integration and testing in a high-fidelity trajectory simulation.

(7) Test the abort protocol developed here on vehicles that have no throttle capability. Although most reusable launch vehicle concepts do have at least some minimal amount of throttling capability, it would be interesting to see if abort performance is adequate for fixed-thrust vehicles in the worst-case scenario.

(8) Modularize and clean up the code developed in this effort so that it is easier to use and understand. This will be beneficial in providing to other organizations that have a use for such algorithms and make it easier to port to other computers, perhaps for testing efforts. Such an effort would also make it easier to maintain and add capabilities to the code.

APPENDIX A

MISCELLANEOUS GUIDANCE RELATIONS

This appendix presents miscellaneous relations that went into the guidance algorithm development. In particular, mass and throttle evolution during constant acceleration arcs and predicted time at which a particular "minimum" throttle setting will be attained during a constant acceleration arc. These are critical parts of the predictor portion of the predictor-corrector.

During constant thrust phases, rocket-powered vehicle mass is calculated from the simple relationship:

$$m(t) = m(t_0) + \dot{m}(t - t_0) \tag{A-1}$$

During constant acceleration phases, the vehicle mass expression is obtained as follows. Start with the thrust/force equation:

$$T(t) = -\dot{m}(t)V_{ex} = m(t)\bar{a} \tag{A-2}$$

Manipulate the two right-hand expressions and integrate to get the mass expression

$$\frac{dm(t)}{m(t)} = \frac{-\bar{a}}{V_{ex}} dt$$

$$\ln(m(t)) = \ln(m(t_0)) - \frac{\bar{a}}{V_{ex}}(t - t_0) \tag{A-3}$$

$$m(t) = m(t_0)\exp\left(\frac{-\bar{a}}{V_{ex}}(t - t_0)\right)$$

During an acceleration limited phase, the following relation holds:

$$\eta(t)T_{\max} = \bar{a}m(t) = \bar{a}m(t_0)\exp\left(\frac{-\bar{a}}{V_{ex}}(t - t_0)\right) \tag{A-4}$$

The time at which a 'minimum' throttle value will be attained can be calculated by manipulating the previous as follows:

$$\eta_{\min} = \frac{\bar{a}m(t_0)\exp\left(\frac{-\bar{a}}{V_{ex}}(t_{\min} - t_0)\right)}{T_{\max}}$$

$$t_{\min} = t_0 - \frac{V_{ex}}{\bar{a}}\ln\left(\frac{T_{vac}\eta_{\min}}{\bar{a}m(t_0)}\right) \tag{A-5}$$

The above is used in the guidance in cases where a minimum throttle setting is specified. Typically, upon reaching this point, part of the propulsion system is shut down and the remaining part is throttled up.

APPENDIX B

CONSTRAINT ON FINAL HAMILTONIAN IS IGNORABLE

This appendix addresses the subtle problem regarding whether or not the constraint on the final Hamiltonian, $H(t_f) = 0$ can be ignored and replaced with a simpler constraint, such as fixing one of the components of the initial primer vector or constraining the magnitude of the initial (or final) primer vector to be unity. It turns out that the necessary conditions of optimality, as stated in Chapter 6, ensure that this is the case. This fact is used to simplify the guidance code developed as part of this research.

Theorem: $H(t_f) = 0$ Is Ignorable For the Multi-Arc Trajectory Optimization Problem The classical necessary condition of optimality for the free final time problem, $H(t_f) = 0$, is an ignorable constraint for the multi-arc rocket trajectory optimization problem.

Proof: Suppose that all the necessary conditions of optimality are satisfied on a particular multi-arc trajectory except for possibly the necessary condition $H(t_f) = 0$. This means that over the entire trajectory, the Hamiltonian H is non-zero, i.e.

$$H(t) = H_{NT}(t) + TS(t) \neq 0 \tag{B-1}$$

. Recall that in the multi-arc optimization problem, the necessary condition $H_{NT}(t_1) = 0$ at the first free switching point (engine on/off time), t_1, was derived. Thus, the switching function at t_1, $S(t_1)$, must be non-zero:

$$S(t_1) = \frac{|\boldsymbol{\lambda}_v|}{m} - \frac{\lambda_m}{c} - \frac{1}{c} \neq 0 \tag{B-2}$$

Because mass does not appear in the terminal state constraints, the mass costate is zero at the final time and negative elsewhere, the latter following from the non-negative time rate of change of mass costate:

$$\dot{\lambda}_m(t) = \frac{|\boldsymbol{\lambda}_v|T}{m^2} \geq 0, \quad \forall t \tag{B-3}$$

From the well-known linear, homogeneous property of the costates (for the rocket trajectory optimization problem), every component of the initial costate vector (including mass costate) can be scaled (by a positive factor) without disturbing the trajectory or satisfaction of any of the necessary conditions. The only apparent effect of initial costate scaling is a scaling of the costate history:

$$\boldsymbol{\lambda}(t_0) \mapsto \boldsymbol{\lambda}(t) \Rightarrow k\boldsymbol{\lambda}(t_0) \mapsto k\boldsymbol{\lambda}(t), \qquad k > 0 \tag{B-4}$$

Examining the expression for the switching function, it is seen that $S(t_1) = 0$ can be obtained by scaling the initial costate vector by the factor k given by:

$$k = \frac{1/c}{|\boldsymbol{\lambda}_v|/m - \lambda_m/c} > 0 \tag{B-5}$$

But $H_{NT}(t_1) = 0$, $S(t_1) = 0$ means $H(t_1) = 0$ which implies that $H(t_f) = 0$. Therefore, for the multi-arc optimization problem the constraint $H(t_f) = 0$ can be ignored.

——

Ironically, a mathematically rigorous proof for the case of single arc trajectory optimization is, as of the time of this writing, unavailable. (Note that in the special, and common, case wherein $H_{NT}(t_f) = 0$, then, clearly, there exists a positive scalar k to make $H(t_f)$ equal zero. See Appendix C for conditions that guarantee that $H_{NT}(t_f) = 0$) The key to the multi-arc proof is the necessary condition $H_{NT}(t_1) = 0$. The latter is not a necessary condition for the single burn arc (no preceding coast arc) optimization problem.

The theorem proves that there does indeed exist a positive scaling factor k enabling trivial satisfaction of $H(t_f) = 0$ for any given multi-arc trajectory that satisfies all of the other necessary conditions of optimality. No mathematically rigorous justification for ignoring the $H(t_f) = 0$ constraint was found in a review of the classical works of Breakwell [50], Lawden [51], Pontryagin [52], nor in the works of many other researchers. From a practical viewpoint, if a numerically computed trajectory 'looks' optimal and reasonable, then it makes sense to declare victory and proceed without concerning ourselves with the remote possibility that a better trajectory could have been obtained if only $H(t_f)$ had been strictly enforced to be zero. Moreover, there is also the persuasive argument that the initial costate (6-) vector only contains 5 meaningful degrees of freedom and hence at least one of the necessary conditions is, accordingly, trivial, the obvious one being the constraint on $H(t_f)$.

Treating subject constraint as ignorable (non-trivially) simplifies the code, and simplicity is a primary goal in the design of guidance algorithms. Extensive numerical experience strongly indicates that the subject constraint can be safely ignored, at least for the class of problems that have been investigated in this thesis. The primary reason this seemingly trivial detail is addressed at all here is for the possibility of increased understanding of the problem that mathematical analysis often provides. An additional reason is that the question is still being raised periodically by a current leading researcher in this field.[53]

APPENDIX C

CONDITIONS UNDER WHICH $H_{NT}(T_F)$ IS GUARANTEED TO VANISH

In this appendix, conditions are provided that guarantee that $H_{NT}(t_f) = 0$ on an extremal trajectory. This helps us to understand the results of Chapter 6 where this result was observed numerically even though no such constraint was explicitly imposed on the problem.

Previously, it was proven [46] that when all of the k prescribed terminal state constraints are 'Keplerian constants', i.e.,

$$\dot{\psi}_i(x) = 0, \qquad i = 1,2,\ldots,k \tag{C-1}$$

evaluated in the absence of thrust, and if all the transversality conditions are satisfied, then $H_{NT}(t_f)$ is guaranteed to be zero, whether or not $H_{NT}(t_f) = 0$ is explicitly constrained. The following statement expands the conditions under which this guarantee holds.

Theorem: Conditions Under Which $H_{NT}(t_f) = 0$ Is Guaranteed Suppose that all of the k prescribed terminal state constraints satisfy, at the final time t_f, the relations

$$\dot{\psi}_i\left(x\left(t_f\right)\right) = 0, \qquad i = 1,2,\ldots,k \tag{C-2}$$

evaluated in the absence of thrust, and that all of the transversality conditions are satisfied. Then $H_{NT}(t_f)$ is guaranteed to be zero, whether or not $H_{NT}(t_f) = 0$ is explicitly enforced.

Proof:

$$H_{NT} = \boldsymbol{\lambda}^T \begin{bmatrix} v \\ g \end{bmatrix} = \mathbf{v}^T \boldsymbol{\psi}_x \begin{bmatrix} \mathbf{v} \\ \mathbf{g} \end{bmatrix} = \mathbf{v}^T \dot{\boldsymbol{\psi}} = \mathbf{v}^T 0 = 0 \tag{C-3}$$

——

Note that this theorem merely requires that the time derivatives of the constraints be zero at the final time, not for all time thereafter. For example, if radius magnitude, speed and flight path angle (= 0) constraints are imposed consistent with a target circular orbit, then, when these terminal constraints are satisfied, the hypotheses of the theorem are satisfied but not the previous hypotheses requiring the terminal state constraints to be Keplerian constants (note that the radius magnitude constraint is not a 'Keplerian' constraint). Hence, for missions to a circular orbit, $H_{NT}(t_f) = 0$ should always be observed even when the explicitly prescribed constraints are not Keplerian constants.

It is very common for prescribed terminal constraint sets to satisfy the hypothesis of this theorem. Any mission in which all the target conditions can be described in terms of Keplerian quantities, even if the explicit terminal constraints are not Keplerian, satisfies the hypotheses of the theorem. Are there any useful constraint sets that do not satisfy the hypotheses of the theorem? Yes, e.g., the constraint set that includes radius, speed, and flight path angle whenever the target values don't correspond to a circle.

APPENDIX D

PARTIAL DERIVATIVES EXPRESSIONS

This appendix describes the general techniques used to obtain the Jacobian matrix used in the modified Newton's method. The techniques described here result in an exact, to within machine precision, Jacobian. The Jacobian matrix is composed of the partial derivatives of the terminal state constraints with respect to the initial costates and switching times. Although the differential equations of motion and Euler-Lagrange equations are fairly complex, especially with the incorporation of aerodynamic terms, variational principles can be used to obtain the required expressions. Advantages of this approach include efficient Jacobian computation (relative to a finite difference approach), more progress toward the solution for each iteration. The computational efficiency results because many of the sub-expressions required in the evaluation of the equations of motion and Euler-Lagrange equations are also required to evaluate the variational differential equations. Increased accuracy in the Jacobian helps to ensure the theoretically predicted quadratic convergence of the Newton method-based solution process.

First, the variational system of differential equations are obtained. Two general principles are used here, namely, the well-known chain rule and Leibnez' rule for the differentiation of an integral

$$\frac{d}{dc}\int_p^q f(x,c)dx = \int_p^q \frac{\partial}{\partial c} f(x,c)dx + f(q,c)\frac{dq}{dc} - f(p,c)\frac{dp}{dc} \tag{D-1}$$

The *k* terminal state constraints to be enforced are functions of state **x** and costate **λ** which in turn are functions of the guessed initial costate and guessed final time. The Jacobian is given by:

$$\begin{aligned}\mathbf{J} &= \left[\frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{\lambda}(t_0)}, \quad \frac{\partial \boldsymbol{\psi}}{\partial t_f}\right] \\ &= \left[\frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{\lambda}(t_f)}\frac{\partial \boldsymbol{\lambda}(t_f)}{\partial \boldsymbol{\lambda}(t_0)} + \frac{\partial \boldsymbol{\psi}}{\partial \mathbf{x}(t_f)}\frac{\partial \mathbf{x}(t_f)}{\partial \boldsymbol{\lambda}(t_0)}, \quad \frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{\lambda}(t_f)}\frac{\partial \boldsymbol{\lambda}(t_f)}{\partial t_f} + \frac{\partial \boldsymbol{\psi}}{\partial \mathbf{x}(t_f)}\frac{\partial \mathbf{x}(t_f)}{\partial t_f}\right] \\ &= \left[\frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{\lambda}(t_f)}\frac{\partial \boldsymbol{\lambda}(t_f)}{\partial \boldsymbol{\lambda}(t_0)} + \frac{\partial \boldsymbol{\psi}}{\partial \mathbf{x}(t_f)}\frac{\partial \mathbf{x}(t_f)}{\partial \boldsymbol{\lambda}(t_0)}, \quad \frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{\lambda}(t_f)}\dot{\boldsymbol{\lambda}}(t_f) + \frac{\partial \boldsymbol{\psi}}{\partial \mathbf{x}(t_f)}\dot{\mathbf{x}}(t_f)\right]\end{aligned} \tag{D-2}$$

At any time *t*, state and costate are given by

$$\begin{aligned}\mathbf{x}(t) &= \mathbf{x}(t_0) + \int_{t_0}^{t} \dot{\mathbf{x}}\,dt \\ \boldsymbol{\lambda}(\mathrm{t}) &= \boldsymbol{\lambda}(\mathrm{t}_0) + \int_{t_0}^{t} \dot{\boldsymbol{\lambda}}\,dt\end{aligned} \tag{D-3}$$

so that

$$\begin{aligned}\frac{\partial \mathbf{x}(\mathrm{t})}{\partial \boldsymbol{\lambda}(\mathrm{t}_0)} &= \int_{t_0}^{t}\left(\frac{\partial \dot{\mathbf{x}}}{\partial \mathbf{x}}\frac{\partial \mathbf{x}}{\partial \boldsymbol{\lambda}(\mathrm{t}_0)} + \frac{\partial \dot{\mathbf{x}}}{\partial \boldsymbol{\lambda}}\frac{\partial \boldsymbol{\lambda}}{\partial \boldsymbol{\lambda}(\mathrm{t}_0)}\right)dt \\ \frac{\partial \boldsymbol{\lambda}(\mathrm{t})}{\partial \boldsymbol{\lambda}(\mathrm{t}_0)} &= \mathbf{I}_6 + \int_{t_0}^{t}\left(\frac{\partial \dot{\boldsymbol{\lambda}}}{\partial \mathbf{x}}\frac{\partial \mathbf{x}}{\partial \boldsymbol{\lambda}(\mathrm{t}_0)} + \frac{\partial \dot{\boldsymbol{\lambda}}}{\partial \boldsymbol{\lambda}}\frac{\partial \boldsymbol{\lambda}}{\partial \boldsymbol{\lambda}(\mathrm{t}_0)}\right)dt\end{aligned} \tag{D-4}$$

where $\mathbf{I}_6$ is the 6×6 identity matrix. These equations are often referred to as variational equations. For the sake of brevity, the expressions are not expanded any further for they are derived easily enough using the state-costate equations given in Chapter 2. The variational system is computed (via numerical integration) simultaneously with the state-costate equations. There are a total of 72 (12 (6 states plus 6 costates) times 6 (6 initial costates)) differential variational equations. This sounds like a lot but there are many common sub-expressions such that the extra effort required to evaluate the variational equations is incrementally more than that required to evaluate the state/costate equations.

APPENDIX E

VEHICLE PROPERTIES

This appendix presents the mass, aerodynamic and propulsive properties of the launch vehicles used to test the formulations developed in this thesis. The first vehicle is the single-stage-to-orbit X-33 flight demonstrator [4], while the second is a generic two-stage-to-orbit vehicle.

X-33 Vehicle Characteristics

Single-Stage-To-Orbit, LOX/LH2

Four linear aerospike engines:

T(vacuum) = 267,000 lbs each

T(sea level) = 205,000 lbs each

I_{sp}(vacuum) = 432.5 sec

I_{sp}(sea level) = 338.4 sec

A_e = N/A for aerospike engine. Pressure loss term part of nonlinear data.

Aerodynamic ref. area = 1,608 sq ft

Gross Lift-off Weight	272,700 lbm
Available propellant	189,700 lbm
Maximum acceleration	4.0 g's

Maximum dynamic pressure limit 618 psf

Maximum $q\alpha$ 1,000 psf-deg

Table E-1: X-33 Composite Mass Properties vs. Propellant Remaining

fuel remaining, lbm	mass, lbm	x_{cg}, in	y_{cg}, in	z_{cg}, in
0	83000	500.56	0.000E+00	7.250E+00
4742	87742	496.99	0.000E+00	7.350E+00
9485	92485	491.08	0.000E+00	7.850E+00
18970	101970	479.26	0.000E+00	8.840E+00
47425	130425	451.35	0.000E+00	1.077E+01
94850	177850	412.94	0.000E+00	1.275E+01
142275	225275	381.04	0.000E+00	1.360E+01
189700	272700	350.65	0.000E+00	1.393E+01

Table E-2: X-33 Axial Force Coefficient vs. Mach, α

α/M	0.00	0.30	0.60	0.80	0.90	0.95	1.05	1.15
-4	0.085	0.085	0.0989	0.1223	0.171	0.2098	0.2635	0.2667
-2	0.0943	0.0943	0.0963	0.1205	0.1715	0.2096	0.2651	0.2657
0	0.098	0.098	0.0965	0.1201	0.1702	0.2024	0.2667	0.2652
2	0.0963	0.0963	0.0954	0.1227	0.1674	0.198	0.2647	0.2643
4	0.0903	0.0903	0.0899	0.1218	0.16	0.1961	0.2618	0.2627
6	0.0789	0.0789	0.0813	0.1186	0.1576	0.1912	0.2565	0.2602
8	0.0637	0.0637	0.0692	0.1124	0.1573	0.1842	0.2501	0.2563
10	0.0448	0.0448	0.0551	0.1039	0.1532	0.1793	0.2444	0.2521
12	0.0227	0.0227	0.0426	0.0954	0.148	0.1739	0.2387	0.2483
14	-0.0031	-0.0031	0.0371	0.0868	0.143	0.1669	0.2332	0.2456
16	-0.0277	-0.0277	0.0366	0.0802	0.1391	0.1593	0.2283	0.2437
18	-0.0277	-0.0277	0.0366	0.0802	0.1391	0.1593	0.2283	0.2437
20	-0.0277	-0.0277	0.0366	0.0802	0.1391	0.1593	0.2283	0.2437
22	-0.0277	-0.0277	0.0366	0.0802	0.1391	0.1593	0.2283	0.2437

Table E-2: X-33 Axial Force Coefficient vs. Mach, α: Concluded

α/M	1.46	2.01	2.75	3.49	4.75	6.00	10.00	20.00
-4	0.2421	0.212	0.1807	0.1741	0.1472	0.1522	0.1521	0.1526
-2	0.24	0.2083	0.178	0.1713	0.1436	0.1482	0.1482	0.1483
0	0.2374	0.2052	0.1752	0.1685	0.1402	0.1443	0.1443	0.1443
2	0.2358	0.203	0.1726	0.1662	0.1377	0.1414	0.1417	0.1415
4	0.2332	0.2006	0.1705	0.1642	0.1354	0.1397	0.1404	0.14
6	0.2312	0.199	0.1687	0.1623	0.1337	0.1385	0.1398	0.1389
8	0.2303	0.1983	0.1672	0.1603	0.1322	0.1375	0.1396	0.1381
10	0.2287	0.1966	0.1654	0.1583	0.1309	0.1368	0.1399	0.1376
12	0.2263	0.1954	0.1631	0.156	0.13	0.1363	0.1405	0.1374
14	0.2249	0.1944	0.1611	0.1542	0.1291	0.1357	0.1411	0.1378
16	0.224	0.1915	0.1587	0.1523	0.1285	0.1352	0.1417	0.1392
18	0.224	0.1874	0.1565	0.1501	0.1274	0.1339	0.1418	0.1412
20	0.224	0.1854	0.1546	0.148	0.1266	0.1337	0.1431	0.1442
22	0.224	0.1831	0.152	0.1452	0.1258	0.1339	0.146	0.1474

Table E-3: X-33 Normal Force Coefficient vs. Alpha, Mach

α/M	0.0000	0.3000	0.6000	0.8000	0.9000	0.9500	1.0500	1.1500
-4	-0.2362	-0.2362	-0.2228	-0.2461	-0.2868	-0.2999	-0.2682	-0.2550
-2	-0.1290	-0.1290	-0.1198	-0.1451	-0.1754	-0.1754	-0.1364	-0.1331
0	-0.0158	-0.0158	-0.0099	-0.0367	-0.0576	-0.0426	-0.0053	-0.0128
2	0.0989	0.0989	0.1047	0.0736	0.0719	0.0945	0.1309	0.1099
4	0.2118	0.2118	0.2146	0.1827	0.1902	0.2168	0.2626	0.2326
6	0.3241	0.3241	0.3242	0.2886	0.2883	0.3176	0.3839	0.3537
8	0.4342	0.4342	0.4313	0.3893	0.3874	0.4179	0.4980	0.4707
10	0.5424	0.5424	0.5328	0.4864	0.4857	0.5190	0.6078	0.5854
12	0.6465	0.6465	0.6298	0.5816	0.5712	0.6129	0.7083	0.6940
14	0.7440	0.7440	0.7197	0.6758	0.6543	0.6949	0.7995	0.7960
16	0.8257	0.8257	0.8048	0.7698	0.7250	0.7672	0.8823	0.8932
18	0.8257	0.8257	0.8048	0.7698	0.7250	0.7672	0.8823	0.8932
20	0.8257	0.8257	0.8048	0.7698	0.7250	0.7672	0.8823	0.8932
22	0.8257	0.8257	0.8048	0.7698	0.7250	0.7672	0.8823	0.8932

Table E-3: X-33 Normal Force Coefficient vs. Alpha, Mach: Concluded

α/M	1.46	2.01	2.75	3.49	4.75	6.00	10.00	20.00
-4	-0.2215	-0.2164	-0.1772	-0.2056	-0.1616	-0.1570	-0.1582	-0.1518
-2	-0.1235	-0.1416	-0.1217	-0.1566	-0.1222	-0.1191	-0.1209	-0.1172
0	-0.0239	-0.0645	-0.0644	-0.1059	-0.0834	-0.0815	-0.0858	-0.0826
2	0.0743	0.0124	-0.0061	-0.0553	-0.0448	-0.0441	-0.0519	-0.0474
4	0.1734	0.0899	0.0531	-0.0032	-0.0042	-0.0066	-0.0171	-0.0107
6	0.2732	0.1692	0.1132	0.0506	0.0378	0.0324	0.0197	0.0279
8	0.3715	0.2484	0.1754	0.1066	0.0820	0.0737	0.0587	0.0692
10	0.4669	0.3271	0.2383	0.1649	0.1292	0.1177	0.1004	0.1134
12	0.5614	0.4051	0.3026	0.2241	0.1789	0.1655	0.1457	0.1601
14	0.6548	0.4814	0.3679	0.2873	0.2321	0.2169	0.1952	0.2085
16	0.7476	0.5584	0.4343	0.3520	0.2883	0.2723	0.2496	0.2581
18	0.7476	0.6350	0.5020	0.4188	0.3477	0.3318	0.3076	0.3095
20	0.7476	0.7099	0.5711	0.4871	0.4101	0.3943	0.3684	0.3655
22	0.7476	0.7851	0.6410	0.5564	0.4749	0.4609	0.4310	0.4275

Generic RLV Characteristics

Two-stage Serial burn, LOX/RP booster, LOX/LH2 orbiter

Five LOX/RP booster engines, three LOX/LH2 orbiter engines

Booster engines:

T(vacuum) = 1,260,000 lbs

T(sea level) = 1,183,635 lbs

I_{sp}(vacuum) = 330 sec

I_{sp}(sea level) = 310 sec

A_e = 35.92 sq ft

Orbiter engines:

T = 380,000 lbs

I_{sp} = 450 sec

A_e = 43.731 sq ft

OMS Engine (1):

I_{sp} = 315 sec

T = 25,000 lbs

Booster/Orbiter Configuration aerodynamic ref. area = 7,900 sq ft

Orbiter aerodynamic ref. area = 2,900 sq ft

Gross Lift-off Weight	4,516,272 lbm (to 28.5 deg)	4,500,000 lbm (to 51.6 deg)
Booster available propellant	2,991,000 lbm	
Booster jettison weight	570,000 lbm	

OMS available propellant	7,365 lbm
Usable orbiter propellant	685,000 lbm
Maximum acceleration	3.0 g's
Maximum dynamic pressure limit	650 psf

Table E-4: Generic RLV 1st Stage Composite Mass Properties vs. Propellant Remaining

fuel remaining, lbm	mass, lbm	x_{cg}, ft	y_{cg}, ft	z_{cg}, ft
685000.0	1525272.0	107.0	0.0	-9.0
3676000.0	4516272.0	123.0	0.0	-19.0

Table E-5: Generic RLV 2nd Stage Composite Mass Properties vs. Propellant Remaining

fuel remaining, lbm	mass, lbm	x_{cg}, ft	y_{cg}, ft	z_{cg}, ft
0.0	270272.0	108.0	0.0	4.0
685000.0	955272.0	64.0	0.0	1.0

Table E-6: Generic RLV Lift Coefficient vs. Mach, α

C_L							
M/α	10.000	8.0000	-6.0000	-4.0000	2.0000	0.0000	2.0000
0.3000	0.3317	0.1803	-0.1109	-0.0424	0.0254	0.0938	0.1630
0.6000	0.3665	0.1908	-0.1158	-0.0439	0.0272	0.0989	0.1715
0.9000	0.4247	0.2229	-0.1313	-0.0484	0.0314	0.1121	0.1938
1.1000	0.5012	0.2688	-0.1617	-0.0606	0.0348	0.1301	0.2272
1.3000	0.4356	0.2406	-0.1496	-0.0634	0.0203	0.1033	0.1869
1.4600	0.3877	0.2176	-0.1372	-0.0603	0.0149	0.0902	0.1665
1.9600	0.3339	0.1967	-0.1285	-0.0611	0.0015	0.0655	0.1253
2.7400	0.2430	0.1463	-0.0972	-0.0499	0.0079	0.0393	0.0828
2.9900	0.2260	0.1365	-0.0913	-0.0479	0.0091	0.0344	0.0748
3.4800	0.2010	0.1220	-0.0826	-0.0449	0.0110	0.0272	0.0630
4.0000	0.1818	0.1108	-0.0759	-0.0425	0.0123	0.0216	0.0539
4.4500	0.1693	0.1034	-0.0715	-0.0409	0.0132	0.0179	0.0479
4.9600	0.1579	0.0966	-0.0673	-0.0393	0.0140	0.0146	0.0425
10.0000	0.1137	0.0696	-0.0507	-0.0330	0.0169	0.0014	0.0209
20.0000	0.0981	0.0588	-0.0433	-0.0299	0.0178	0.0040	0.0124
30.0000	0.0968	0.0572	-0.0421	-0.0295	0.0184	0.0059	0.0097

Table E-6: Generic RLV Lift Coefficient vs. Mach, α: Concluded

C_L								
M/α	4.000	6.000	8.000	10.00	12.00	14.00	16.000	20.000
0.300	0.2328	0.3747	0.5652	0.7701	0.9772	1.179	1.3682	1.5374
0.600	0.2448	0.3963	0.6025	0.8181	1.0348	1.2452	1.4419	1.6173
0.900	0.2767	0.4510	0.6843	0.9259	1.1684	1.4040	1.6249	1.8232
1.100	0.3260	0.5321	0.8047	1.0881	1.3757	1.6608	1.9363	2.1949
1.300	0.2728	0.4505	0.6831	0.9240	1.1691	1.4141	1.6544	1.8849
1.460	0.2446	0.4056	0.6150	0.8318	1.0520	1.2744	1.4965	1.7155
1.960	0.1871	0.3208	0.4990	0.6842	0.8737	1.0547	1.2219	1.3738
2.740	0.1288	0.2301	0.3697	0.5204	0.6763	0.8291	0.9730	1.0974
2.990	0.1177	0.2130	0.3462	0.4906	0.6419	0.7921	0.9320	1.0536
3.480	0.1015	0.1881	0.3114	0.4475	0.5929	0.7384	0.8751	0.9945
4.000	0.0889	0.1688	0.2850	0.4161	0.5576	0.7004	0.8355	0.9540
4.450	0.0806	0.1563	0.2679	0.3960	0.5354	0.6767	0.8110	0.9291
4.960	0.0732	0.1449	0.2526	0.3781	0.5152	0.6552	0.7888	0.9066
10.00	0.0438	0.1016	0.1972	0.3141	0.4453	0.5817	0.7135	0.8312
20.00	0.0331	0.0877	0.1760	0.2803	0.4030	0.5323	0.6594	0.7763
30.00	0.0300	0.0854	0.1806	0.2990	0.4328	0.5719	0.7058	0.8240

Table E-7: Generic RLV Drag Coefficient vs. Mach, α

C_D							
M/α	-10.00	-8.000	-6.000	-4.000	-2.000	0.000	2.000
0.3000	0.0644	0.0386	0.0330	0.0302	0.0301	0.0327	0.0380
0.6000	0.0869	0.0463	0.0399	0.0370	0.0370	0.0399	0.0457
0.9000	0.1176	0.0683	0.0563	0.0542	0.0539	0.0579	0.0651
1.1000	0.2129	0.1495	0.1326	0.1245	0.1247	0.1306	0.1433
1.3000	0.2098	0.1518	0.1356	0.1271	0.1253	0.1302	0.1414
1.4600	0.1952	0.1474	0.1342	0.1274	0.1265	0.1313	0.1415
1.9600	0.1601	0.1071	0.0871	0.0766	0.0726	0.0747	0.0833
2.7400	0.1354	0.0937	0.0802	0.0718	0.0678	0.0686	0.0741
2.9900	0.1310	0.0915	0.0790	0.0710	0.0677	0.0676	0.0723
3.4800	0.1250	0.0891	0.0776	0.0703	0.0667	0.0665	0.0703
4.0000	0.1196	0.0864	0.0757	0.0687	0.0651	0.0647	0.0678
4.4500	0.1173	0.0856	0.0754	0.0686	0.0651	0.0644	0.0673
4.9600	0.1183	0.0885	0.0791	0.0726	0.0691	0.0681	0.0705
10.0000	0.1086	0.0774	0.0644	0.0579	0.0542	0.0534	0.0567
20.0000	0.1074	0.0787	0.0693	0.0629	0.0586	0.0565	0.0576
30.0000	0.1124	0.0908	0.0850	0.0811	0.0786	0.0767	0.0769

Table E-7: Generic RLV Drag Coefficient vs. Mach, α: Concluded

C_D								
M/α	4.00	6.00	8.00	10.00	12.00	14.00	16.00	20.00
0.30	0.0463	0.0719	0.1449	0.2815	0.4756	0.7277	1.0376	1.4036
0.60	0.0546	0.0868	0.1796	0.3305	0.5398	0.8091	1.1381	1.5250
0.90	0.0761	0.1206	0.2292	0.3971	0.6271	0.9209	1.2785	1.6979
1.10	0.1617	0.2244	0.3614	0.5673	0.8455	1.1986	1.6278	2.1324
1.30	0.1593	0.2180	0.3412	0.5222	0.7641	1.0692	1.4385	1.8722
1.46	0.1591	0.2138	0.3229	0.4798	0.6861	0.9439	1.2546	1.6189
1.96	0.0985	0.1489	0.2546	0.4125	0.6298	0.9059	1.2447	1.6532
2.74	0.0842	0.1209	0.2037	0.3259	0.4996	0.7271	1.0027	1.3337
2.99	0.0816	0.1158	0.1919	0.3105	0.4775	0.6954	0.9643	1.2847
3.48	0.0784	0.1089	0.1787	0.2892	0.4470	0.6547	0.9125	1.2187
4.00	0.0749	0.1028	0.1679	0.2731	0.4247	0.6258	0.8764	1.1743
4.45	0.0738	0.0999	0.1623	0.2643	0.4123	0.6096	0.8562	1.1491
4.96	0.0765	0.1016	0.1616	0.2617	0.4073	0.6015	0.8446	1.1337
10.00	0.0636	0.0931	0.1520	0.2485	0.3887	0.5753	0.8083	1.0848
20.00	0.0619	0.0886	0.1749	0.3100	0.4589	0.6468	0.8747	1.1390
30.00	0.0798	0.0964	0.1449	0.2313	0.3631	0.5441	0.7750	1.0522

APPENDIX F

TREATMENT OF GIMBALLED ENGINE VEHICLES

This appendix describes a method for handling the common case of axially asymmetric launch vehicles. The *z*-body axis component of the center of gravity of these vehicles varies as a function of total vehicle mass. This variation requires that the engines be gimballed such that the net propulsive force acts through the vehicle center of gravity to prevent unsteady pitching moments. This gimballing results in a *z*-body component of thrust, which, if not accounted for in the atmospheric equations of motion part of the optimization process, results in large trajectory deviations.

The equations of motion for a thrusting rocket in atmospheric flight are:

$$\begin{aligned} \dot{\mathbf{r}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= \mathbf{g} + \frac{(T_x - A)\mathbf{x_b} + (T_z - N)\mathbf{z_b}}{m} \end{aligned} \tag{F-1}$$

where the thrust components, *Tx* and *Tz*, and the axial and normal forces, *A* and *N* are given by:

$$\begin{aligned} T_x &= T\cos\delta_p \\ T_z &= T\sin\delta_p \\ T &= T_{vac} + \Delta T(h) \\ A &= qSC_A \qquad N = qSC_N \end{aligned} \tag{F-2}$$

and δ_p is the pitch gimbal angle, that is, the angle measured in the vehicle's *x*-*z*-body axes plane, from the net thrust vector to the *x*-body axis. The gimbal angle modeled here is the angle required to ensure that the net thrust vector goes through the c.g., hence nulling the net pitching moment acting on the vehicle. Figure F-1 shows a typical plot of gimbal angle as a function of mass during first-stage flight. Note that it is nearly a linear function of vehicle mass so it is modeled by:

$$\delta_p = a_0 + a_1 m \tag{F-3}$$

where a_0, a_1 are coefficients obtained offline with a least squares curve fit. The effect of neglecting the gimbal angle in the optimization is severely degraded performance, especially for vehicles with large gimbal angles such as that depicted in Figure F-1. If gimbal angle effects were to be ignored, the guidance would command, for example, an inertial-to-body quaternion, and the attitude control system would indeed maneuver the body axes to the commanded orientation but the engine nozzles would be gimbaled in such a way as to effectively null total pitching moment. The net thrust vector direction, then, would vary from that modeled by guidance, resulting in sub-optimal performance.

Note that the effects of aerodynamically-induced moments are neglected in the guidance model because they are short-lived (i.e., only significant during the high dynamic pressure portion of flight) and cause perturbations of only, say, 1 or 2 degrees in the net thrust vector direction, thus, performance is not significantly affected by neglecting them.

During second stage flight, the dynamic pressure is typically sufficiently small that the guidance does not need to model gimbal angle effects when solving the boundary value problem. This is because, when dynamic pressure, and hence, aerodynamic accelerations are low, so that

most of the contact acceleration is due to thrust, the (effective) pitch gimbal angle can be accurately estimated from navigation information:

$$\delta p_{est} = \tan^{-1}\left(\frac{a_z}{a_x}\right) \tag{F-4}$$

where a_z and a_x are the sensed accelerations along the vehicle's x- and z-body axes. The guidance then biases the pitch angle resulting from the boundary value problem solution by the estimated gimbal angle so that when the attitude control system orients the body axes to the commanded orientation, the net thrust vector will be in the direction in which guidance calculates it needs to be in order to attain the target conditions.

The following describes the optimality condition, which is modified from that of Chapter 2 because of the thrust component along the z-body axis. Applying the maximum principle to the Hamiltonian results in the following optimization sub-problem:

$$\max_{\mathbf{x_b}}\left\{\boldsymbol{\lambda}_v{}^T\left[\frac{(T_x - A)\mathbf{x_b} + (T_z - N)\mathbf{z_b}}{m}\right]\right\} \tag{F-5}$$

Note that the optimal control, $\mathbf{x_b}^{\mathbf{o}}$ (and hence, optimal z-body axis, $\mathbf{z_b}^{\mathbf{o}}$) lies in the plane defined by the (initial) position and velocity-costate vectors [46]. Thus, the optimization sub-problem (F-5) can be written simply as

$$\max_{\delta}\{(T_x - A)\cos(\delta) + (T_z - N)\sin(\delta)\} \tag{F-6}$$

where δ is the angle between the x-body axis and the primer vector.

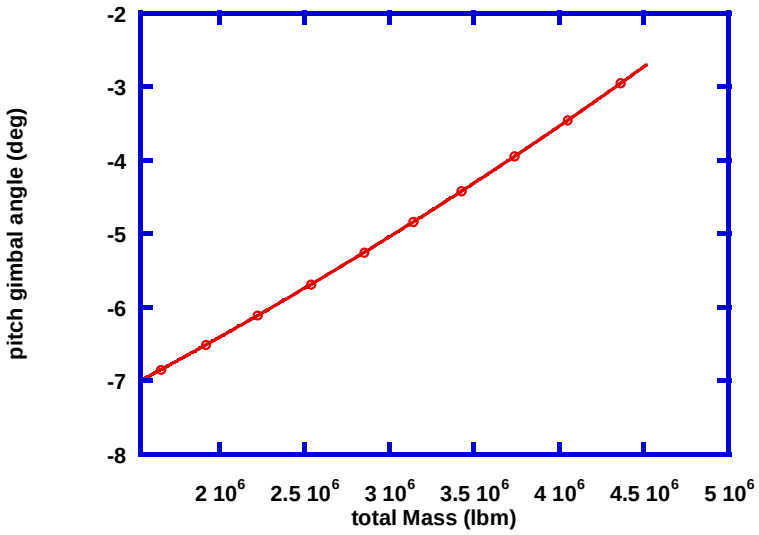

Figure F-1: Typical First Stage Pitch Gimbal Angle

REFERENCES

[1] Anon., "The Space Launch Initiative: Technology to Pioneer the Space Frontier," NASA Marshall Space Flgith Center, Pub. 8-1250, FS-2001-06-122-MSFC, June, 2001.

[2] Hanson, John, "Advanced Guidance and Control Project for Reusable Launch Vehicles," AIAA-2000-3957 Presented at the 2000 AIAA Guidance, Navigation, and Control Conference, Denver, CO, August 2000.

[3] McHenry, R. L., Brand, T. J., Long, A. D., Cockrell, B. F., and Thibodeau, J. R. III, "Space Shuttle Ascent Guidance, Navigation, and Control," *Journal of the Astronautical Sciences*, Vol. XXVII, No. 1, pp. 1-38, January-March, 1979.

[4] Hanson, J. M., Coughlin, D. J., Dukeman, G. A., Mulqueen, J. A., McCarter, J. W., "Ascent, Transition, Entry, and Abort Guidance Algorithm Design for the X-33 Vehicle," AIAA Paper 98-4409, 1998.

[5] Smith, I.E., "General Formulation of the Iterative Guidance Mode," NASA TM X-53414 (corrected copy), NASA George C. Marshall Space Flight Center, Huntsville, AL, March 22, 1966.

[6] Chandler, D. C., Smith, I. E., "Development of the Iterative Guidance Mode with its Application to Various Vehicles and Missions," *J. Spacecraft*, vol. 4, p. 898-903, 1967.

[7] Bryson, Arthur, E., Jr., and Ho, Yu-Chi, *Applied Optimal Control*, Hemisphere Publishing Corporation, 1975, pp. 59-62.

[8] Brown, K.R., Harrold, E.F., and Johnson, G.W., "Rapid Optimization of Multiple-Burn Rocket Trajectories," NASA CR-1430, September 1969.

[9] Cohen, A., O., Brown, K. R., "Real-Time Optimal Guidance for Orbital Maneuvering," *AIAA Journal*, Vol. 11, No. 9, September 1973, pp. 1266-1272.

[10] Burrows, R. R., McDaniel, G. A., "A Method of Trajectory Analysis With Multi-Mission Capability and Guidance Application," AIAA Paper No. 68-844, August 1968.

[11] Sinha, S.K., and Shrivastave, S.K., "Optimal Explicit Guidance of Multistage Launch Vehicle Along Three-Dimensional Trajectory," *Journal of Guidance, Dynamics and Control*, Vol. 13, No. 3, pp. 394-403.

[12] Lu, Ping, "Nonlinear Trajectory Tracking Guidance with Application to a Launch Vehicle," *Journal of Guidance, Control and Dynamics*, Vol. 19, No. 1, pp. 99-106, January-February, 1996.

[13] Seywald, Hans, "Neighboring Optimal Control Based Feedback Law for the Advanced Launch System," *Journal of Guidance, Control and Dynamics*, Vol. 17, No. 6, pp. 1154-1162, November-December 1994.

[14] Brown, K. R., Harrold, E.F., Johnhson, G. W., "Some New Results on Space Shuttle Atmospheric Ascent Optimization," AIAA Paper No. 70-978, 1970.

[15] Ingram, H. L., "Closed-form Solutions for Atmospheric Flight With Applications to Shuttle Guidance," NASA TM X-64694, 1972.

[16] Kelly, W. D., "Formulation of Aerodynamic Quantities for Minimum Hamiltonian Guidance," AIAA Paper 92-4380, 1992.

[17] Dickey, L.R., "Guidance Applications of Linear Analysis," Marshall Space Flight Center, Huntsville, Alabama, TM X-53166, 27 November 1964.

[18] Battelle, C.T., Gottlieb, R.G., "Optimization of Thrust Direction Histories and Vehicle Parameters for Three-Dimensional Ascent Trajectories," *Journal of Spacecraft and Rockets*, Vol. 3, No. 5, May 1966.

[19] Gottlieb, R.G., "Rapid Convergence to Optimum Solutions Using a Min-H Strategy," *AIAA Journal*, Vol. 5, No. 2, pp. 322-329, February 1967.

[20] Bradt, J. E., Jessich, M. V., and Hardtla, J. W., "Optimal Guidance for Future Space Applications," AIAA Paper 87-2401, August 17-29, 1987, Monterey, CA.

[21] Cramer, E. J., Bradt, J. E., Hardtla, J. W., "Launch Flexibility Using NLP Guidance and Remorte Wind Sensing," AIAA Paper 90-3330-CP, 1990.

[22] Hanson, J., Shrader, M., Cruzen, C., "Ascent Guidance Options," AIAA Paper 94-3568-CP, 1994.

[23] Leung, M. S. K., Calise, A. J., "A Hybrid Approach to Near-Optimal Launch Vehicle Guidance," AIAA Paper 92-4304-CP, 1992.

[24] Calise, A. J., Melamed, N., and Lee, Seungjae, "Design and Evaluation of a Three-Dimensional Optimal Ascent Guidance Algorithm," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 6, pp. 867-875, Nov-Dec 1998.

[25] Gath, P., and Calise, A., "Optimization of Launch Vehicle Ascent Trajectories With Path Constraints and Coast Arcs," AIAA Paper 99-4308, August 1999.

[26] Ross, I.M., "Extremal Angle of Attack Over a Singular Thrust Arc in Rocket Flight," *Journal of Guidance, Control and Dynamics*, Vol. 20, No.2, 1997, pp. 391-393.

[27] Casalino, L, "Singular Arcs During Aerocruise," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 1, Jan-Feb 2000, pp. 118-123.

[28] Calise, A., and Brandt, N., "Generation of Launch Vehicle Abort Trajectories Using a Hybrid Optimization Method," *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 5, November-December 2004.

[29] Bown, R.L., "Return-To-Launch-Site Trajectory Shaping," NASA-CR-15100, Oct 17, 1975.

[30] Carter, J.F., Bown, R.L., "Space Shuttle Three Main Engine Return To Launch Site Abort," NASA-CR-150968, Nov 21, 1975.

[31] Sponagle, Steven J., Fernandes, Stanley T., "Space Shuttle Guidance for Multiple Main Engine Failures During First Stage," *Journal of Guidance, Control and Dynamics*, Vol. 12, No. 6, pp. 880-885, 1989.

[32] Kishi, M. W.Gavert, D. E.,Dee, Y. T., "Development And Performance Evaluation Of The Space Shuttle Vehicle Guidance And Control For Return To Launch Site Aborts," AIAA Paper 1980-907, *Proceedings of the American Institute of Aeronautics and Astronautics, International Meeting and Technical Display on Global Technology 2000*, Baltimore, Md., May 6-8, 1980.

[33] Dutton, Kevin E., "Optimal Control Theory Determination of Feasible Return-to-Launch-Site Aborts for the HL-20 Personnel Launch System Vehicle," NASA Langley Research Center NASA TP-3449 , July 1994.

[34] Stanley, D.O., and Powell, R.W., "Abort Capabilities of Rocket-Powered Single-Stage Launch Vehicles," Journal of Spacecraft and Rockets, Vol. 28, No. 2, pp 184-191, 1991.

[35] Chuang, C.-H., and Ledsinger, Laura A., "First And Second Variation Analysis For Return To Launch Site Guidance," AIAA Paper 1996-3426, *Proceedings of the AIAA Atmospheric Flight Mechanics Conference*, San Diego, CA, July 29-31, 1996.

[36] Betts, J.T., "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 193-207.

[37] Vinh, N.X., "Integrals of the Motion for Optimal Trajectories in Atmospheric Flight," *AIAA Journal*, 11, (1973), 700-703.

[38] Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P., *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed., Cambridge University Press, 1992.

[39] Keller, H.B., *Numerical Methods for TwoPoint Boundary-Value Problems*, Blaisdell Publishing Company, Waltham, Massachussetts, 1968.

[40] Stoer, J., Bulirsch, R., *Introduction to Numerical Analysis*, Springer-Verlag, New York, 1980.

[41] Lu, P, Sun, H., Tsai, B., "Closed-Loop Endoatmospheric Ascent Guidance," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 2, pp. 283-294, 2003.

[42] Jezewski, D.J., "Optimal Analytic Multiburn Trajectories," *AIAA Journal*, Vol. 10, No.5, 1972, pp. 680-685.

[43] Azimov, D.M., Bishop, R.H., "Extremal Rocket Motion with Maximum Thrust in a Linear Central Field," *Journal of Spacecraft and Rockets*, Vol. 38, No.5, September-October 2001, pp. 765-776.

[44] Teren, Fred, and Spurlock, Omer F., "Optimal Three Dimensional Launch Vehicle Trajectories with Attitude and Attitude Rate Constraints," Lewis Research Center, NASA Technical Note, NASA TN-D-5117, NASA, Washington, D.C., March 1969.

[45] Barrett, R. R., Cooper, D. L., and Ellis, L. V., "Transversality Conditions for Circular and Non-Circular Orbits," Research and Analysis Section Technical Memorandum No. 197, Northrup Space Laboratories, Huntsville, AL, September 1966.

[46] Dukeman, G., "Atmospheric Ascent Guidance for Rocket-Powered Launch Vehicles," AIAA Paper 2002-4559.

[47] Dukeman, G. and Calise, A., "Enhancements to an Atmospheric Ascent Guidance Algorithm," AIAA Paper 2003-5638.

[48] Lu, P., Shen, Z., Dukeman, G., Hanson, J., "Entry Guidance by Trajectory Regulation," AIAA paper 2000-3958, *Proceedings of Guidance, Navigation and Control Conference*, August 14-17, 2000, Denver, CO.

[49] Klumpp, Allan R., "Apollo Lunar Descent Guidance," *Automatica*, Vol. 10, pp. 133-146, 1974.

[50] Breakwell, J.V., "The Optimization of Trajectories," *Journal of the Society of Industrial and Applied Mathematics*, Vol. 7, No. 2, June 1959, pp. 215-247.

[51] Lawden, D.K., *Optimal Trajectories for Space Navigation*, Butterworth & Co. Ltd., London, 1963.

[52] Pontryagin, L.S., Boltyanskii, R.V., Gamkrelidze, and E.F. Mishchenko, *The Mathematical Theory of Optimal Processes*, Interscience Publishers, Inc., New York, 1962.

[53] Private communications with Professor Ping Lu of Iowa State University who is also actively involved in research into the powered flight guidance problem, Summer 2003.